# Virginia Polytechnic Institute and State University

Electrical Engineering

BLACKSBURG, VIRGINIA 24061

# SIMULATED LUMPED-PARAMETER SYSTEM REDUCED-ORDER ADAPTIVE CONTROL STUDIES

by

C. Richard Johnson, Jr., D. A. Lawrence,
T. Taylor and M. V. Malakooti

March 1981

Technical Report No. EE-8119
Department of Electrical Engineering
Virginia Polytechnic Institute and State University
Blacksburg, VA 24061

---

This work supported by NASA Grant NAG-I-7.
NASA Technical Officer: Dr. R. C. Montgomery, Flight Dynamics and Control Division, NASA Langley Research Center, Hampton, VA 23665.

# I. INTRODUCTION

Very little conclusive insight is currently available for predicting the behavior of reduced-order adaptive controllers (ROAC), i.e. an adaptive controller based on a plant model of lower dimension than that of the actual plant. In [1] the ROAC problem is seen to be an unavoidable consequence of the application of any finite-dimensional, lumped-parameter system (LPS) adaptive control strategy to an infinite dimensional, distributed-parameter system (DPS). In [2] and [3] the relative orders of the plant, model, and controller are noted to be critical. In [4] the problem of adaptive parameterization of a C order controller for a M order model of a N order system is addressed when N = M > C. However, as noted in [2] and [3], the practical, poorly understood case is when N > M $\geq$ C. This condition, i.e. the system order exceeds that of the plant model used for adaptive controller parameterization, will be considered to constitute the ROAC problem in this report.

Though numerous strategies have been espoused [5] for extracting a low order model from a too complex system description, none currently seem fully applicable to the real-time, recursive requirements of on-line adaptive control algorithms. These reduced-order modeling techniques seem to fall into two broad categories: (i) Extract those "modes" (or component-subsystems) from the full system description that are most influential in the performance of the model in its subsequent use. This strategy is followed, for example, in [6] and [7]. (ii) Parameterize the reduced-order structure to provide the best prediction of the desired output. This latter strategy will produce a model the "modes" of which need not correspond to any of those of the full system as noted in [3]. This latter strategy includes the model reference approach prominent in adaptive systems [8]. Two methods of interpreting the misbehavior of ROAC, one

based on system input-output description [3] [9] and one on state-variable description [1] [10], have emerged recently from attempts to develop adaptive controllers for flexible spacecraft.

Consider the implementation of the proper, single-input, single-output (SISO), autoregressive, moving-average (ARMA) system

$$y(k) = \sum_{i=1}^{N} [a_i\; y(k-i) + b_i\; u(k-i)] \quad \text{(N even)}, \tag{1-1}$$

where u is the system input and y the output, in parallel, i.e. partial fraction expanded, form as

$$y_\ell(k) = \sum_{i=1}^{2} [\alpha_{i\ell}\; y_\ell(k-i) + \beta_{i\ell}\; u(k-i)] \tag{1-2}$$

$$y(k) = \sum_{\ell=1}^{N/2} y_\ell(k), \tag{1-3}$$

where $y_\ell$ is the output of the $\ell$-th mode. As interpreted in [3] the first strategy in the preceding paragraph when used for identification would lead to approximation of (1-3) as

$$y_s(k) = \sum_{\ell}^{M/2 \text{ of } [1, N/2]} y_\ell(k) \tag{1-4}$$

with (1-2) describing the dynamics of each $y_\ell(k)$. The $M/2$ modes chosen in (1-4) may be selected as those $M/2$ from the $N/2$ in (1-3) that provide the "best" fit of $y_s$ to y. Note that the order M of (1-4) refers to the order of the underlying reduced-order ARMA model. If the dimension of (1-4) (and (1-1)) were based on the number of quadratic "modes" then M (and N) would be replaced by $\bar{M} = \frac{M}{2}$ (and $\bar{N} = \frac{N}{2}$). This quadratic "modal" designation of order is common in the flexible spacecraft literature [3]. As noted in [1], [3], and [11] the extraction of the $y_\ell$ chosen in (1-4) from the y in

(1-3) remains an open question. However, these $y_\ell$ would be required for identification of the appropriate (1-2) parameters $\alpha_{i\ell}$ and $\beta_{i\ell}$. Alternatively, the second strategy would approximate (1-3) with

$$\hat{y}(k) = \sum_{\ell}^{M/2 \text{ of } [1,\frac{N}{2}]} \hat{y}_\ell(k) \tag{1-5}$$

in, e.g., a least-squares sense by selection of the $\hat{y}_\ell(k)$. These estimated "modal" outputs need not match the corresponding values in (1-2) and, therefore, need not lead to identification of the corresponding $\alpha_{i\ell}$ and $\beta_{i\ell}$ in (1-2) as noted in [3]. The successive $\hat{y}_\ell$ chosen to fit (1-5) to (1-3) may not even obey a time-invariant difference equation of the form of (1-2) with $y_\ell$ replaced by $\hat{y}_\ell$. The problem with closing the adaptive control loop via a simultaneous identification and control strategy could be intensified by feeding back the $\hat{y}_\ell$ instead of the unavailable $y_\ell$ to meet a modal control objective.

The alternate interpretation of the ill-effects of ROAC, derived from the flexible spacecraft control problem [1] [10], begins with the separation of a SISO, state-space model for, e.g., (1-1) into the reduced-order and unmodeled (or residual) segments as

$$\begin{bmatrix} x_N(k+1) \\ x_R(k+1) \end{bmatrix} = \begin{bmatrix} A_N & A_{NR} \\ A_{RN} & A_R \end{bmatrix} \begin{bmatrix} x_N(k) \\ x_R(k) \end{bmatrix} + \begin{bmatrix} b_N \\ b_R \end{bmatrix} u(k) \tag{1-6}$$

$$y(k) = [c_N \quad c_R] \begin{bmatrix} x_N(k) \\ x_R(k) \end{bmatrix} . \tag{1-7}$$

As outlined in [1] the derivation of (1-6) - (1-7) can be viewed as a

projection operation on the full system. From (1-7) the "spillover" of the residual modes into the observation of y(k) via $c_R\, x_R(k)$ and from (1-6) the "spillover" of the control designed for the reduced model into the residual modes $x_R(k+1)$ via $b_R\, u(k)$ are clearly displayed. Also the possible coupling of reduced model modes and residual modes via $A_{NR}$ and $A_{RN}$ is immediately apparent. Shown in [1] and [10] is the predictable fact that if $A_{NR} = \underline{0}$ and $c_R = \underline{0}$ then, assuming the residual modes remain stable, any full-order (N), stable adaptive controller identifying $A_N$, $b_N$, and $c_N$ explicitly or implicitly from y and u alone would retain its stability. If $A_{NR} = \underline{0}$ but $c_R$ and $b_R$ are nonzero then the degradation of ROAC has two sources. Unmodeled components in y via $c_R$ will generate an error that is indistinguishable from parameter error thereby causing adaption. The application of control u to the residual states $x_R$ via $b_R$ will contribute further to this unmodeled component of y. Not only will the parameter estimates be incorrect from use of y and u to identify only $A_N$, $b_N$, and $c_N$ but also the state estimates provided by an adaptive observer will be incorrect, which if fed back could lead to an unpredictable "controlled" response.

These two problems of inappropriate parameter and state estimation are the same ones noted in the first interpretation with $\hat{y}$ approximation of y in (1-5). The ability to extract the $y_\ell$ and obtain $y_s$ corresponds to an effective zeroing of $c_R$. This report will, in part, attempt to implement, compare, and contrast these two strategies embodied in (1-4) and (1-5).

The next section details the specific objectives of this study. Section III presents the example autoregressive, moving-average plants that are to be used in the simulations. Section IV presents the adaptive

control algorithms to be used and their sources in the literature. Section V outlines the formats for the simulated tests including the description of numerical figures of merit to be tabulated in Section VI. Section VII offers interpretations of the test results and Section VIII draws conclusions relevant to the ROAC problem. The last sections of this report include the referenced literature and the appendices including computer program listings.

## II. OBJECTIVES

The principal objective of this study is to test the usefulness of the folklore of reduced-order modelling with respect to adaptive control. In particular four "facts" will be tested:

(i) Heavily damped modes may be neglected relative to more lightly damped modes in reduced-order-model derivation.

(ii) Finite bandwidth actuators limit the number of modes necessary to be modeled.

(iii) An "optimal" reduced-order controller neglects the modes contributing the least degradation in the control system performance measure.

(iv) Indirect and direct adaptive control are essentially equivalent and interchangeable.

The implication to the ROAC problem of each of these statements will be developed in the following paragraphs. The simulations of the following sections will be chosen to test the veracity of these implications. The conclusion of this report will summarize the useful "facts" that either escape unscathed or emerge from these tests.

Since each of these facts has been accepted into the reduced-order and/or adaptive control folklore it is difficult to pinpoint particular references succinctly stating these points. However, several classical control texts contain the source of point (i) in the concept of dominant roots or poles. For example: "The complex conjugate roots near the origin of the s-plane relative to the other roots of the closed-loop system are labeled the dominant roots of the system since they represent or dominate the transient response. The relative dominance of the roots is determined by the ratio of the real parts of the complex roots and

will result in reasonable dominance for ratios exceeding five. ... Dominance .. also depends upon the relative magnitudes ... of the residues evaluated at the complex roots, [which] depend upon the location of the zeros in the s-plane" [12]. Or: "The relative dominance of closed-loop poles is determined by the ratio of the real parts of the closed-loop poles, as well as by the relative magnitudes of the residues evaluated at the closed-loop poles. The magnitudes of the residues depend upon both the closed-loop poles and zeros. If the ratios of the real parts exceed five, and there are no zeros nearby, then the closed-loop poles nearest the $j\omega$ axis will dominate in the transient-response behavior because these poles correspond to transient-response terms which decay slowly" [13, p. 251]. This separation concept has been formalized via singular perturbation theory [14]. Despite the concomitant warning in both [12] and [13] for caution in the use of this rule and the explication of its application to the closed-loop system, point (i) is commonly (though admittedly inappropriately) used for model reduction prior to control design. The (mis)implication for discrete systems is that if the open-loop singularities are separable into a group in the z-plane outside a radius of r (<1) and the other group inside the radius $r^5$ (since $|z|=e^{Re\{s\}T}$, where T is the sample period) then the first group alone provides a highly accurate input-output model of the entire system <u>and</u> therefore provides a useful dimension and parameterization for reduced-order controller design. One objective of this study is to test the usefulness of this guideline for ROAC. Predictably, such a rule will be valid only when the closed-loop system retains its open-loop singularity separability.

A frequently voiced (through unwritten) criticism of the preoccupation with the effects of spillover is based on the assertion that the frequency content of the input to a plant has a strong effect on the most suitable reduced-order model. Since actuators do not have infinite bandwidth and are commonly modeled as essentially low pass filters [15] the higher frequency modes of the system will receive such insignificant excitation that they are ignorable as noted in point (ii). This again extrapolates a reasonable open-loop response mechanism to a closed-loop situation. Again, if the requirement that plant open-loop singularities were shifted only slightly by the feedback were included then this proposition would be strengthened. As such it narrows the more classical concept by utilizing the shape of the input spectrum in addition to the plant singularity constellation. One problem with the use of this idea in the ROAC problem is the nonlinear, time-varying character of the feedback, which does not fit the linear system character of this guideline. That is, during adaption a large, "high frequency" control effort could imbalance the roll-off provided by the low pass actuators. Another problem is the unmodeled phase shift induced by the actuators unless the actuator outputs are available as the identifier inputs. Due to the prevalence of this seemingly untested proposition, construction of a meaningful example for its examination represents another report objective. The influence of input frequency content on reduced-order controller selection will also be tested for various reference signal frequency content distributions.

Points (i) and (ii) rely on the near-equivalence of the two reduced-order modeling strategies interpreted in the introduction: modal selection and full-behavior approximation. The suggestion in point (iii) recognizes the distinction between these two strategies and represents the sensible result of pursuit of the first strategy embodied in (1-4). Also, clearly the modes of the plant may not be as separated as required in the discussion of point (i), either in the open-loop or closed-loop system, which requires a selection mechanism. As shown in [7] for lightly damped systems forced by infinite bandwidth inputs the modal costs are proportional to the product of the modal time constant, observability norm, and disturbability (or controllability) norm. Since for a partial fraction expansion the observability and disturbability norms are related to the modal residual, this suggestion can be viewed as a more sophisticated version of point (i). However, as demonstrated by example in [7] the low frequency modes need not always provide the best open-loop reduced-order model. Again this modal selection procedure is principally intended for an open-loop fit. For reduced-order control application the truncated modes can be viewed as the desired degrees of freedom omitted from the controller. However, a misinterpretation of this procedure would suggest that a reduced-order controller composed of a prespecified number of well-selected modal controllers would be optimal. This would suggest (falsely) that the second ROAC strategy of full approximation in the introduction, as represented by (1-5), could never prove better than the first of selective extraction, as represented by (1-4). The third objective of this study will be to test this implication by comparing single-mode controllers chosen to control single-mode reduced-models of a two mode system and a single-mode controller chosen to control the full system to "match" a single-mode objective. Such a

comparison will clearly rely on the example chosen and is expected, as with the first two objectives, to yield initially ambiguous results. Consider for example a second-order system controlled by a constant output feedback gain to meet a first order response matching the dominant pole of the root locus. The appropriate gain need not be provided by either separate modal controller but a close fit to the objective does exist for some reduced-order controller parameterization.

In the special case of full-order model following, indirect and direct adaptive control have recently been shown to be equivalent [16] [17]. Indirect adaptive control uses the current plant parameter (and state) estimates to solve for the controller parameters (and feedback signal). Direct adaptive control updates the controller parameters such that the control system response matches that of a prescribed model. Discovery of this equivalence has led to highly touted claims of the interchangeability of the algorithms resulting from these two approaches. The implication is that the two approaches are also equivalent for ROAC. Comparison of the indirect self-tuning scheme with the initial plant MA gain known, the direct input matching scheme, and the approximate direct output error identification interpretation with $q_i = f_i$ in [17], all of which could be based on an equation error parameter estimator, does not dispute this claim for the specific model-following problem and equilibrating choice of adaptive gains. If the initial plant MA gain is also estimated then indirect and direct schemes can be expected to behave differently. Only recently has a simple provable direct adaptive control scheme, based on output error parameter estimation, been developed [18]. Like the approximate strategy in [19] a particular choice of designer selected constants reduces this strategy to the common

equation error based scheme in [17]. The complex generation of an additional reference model input required in [20] is avoided in [18].

As an aside, note that even though the reduction of a gradient based solution [21] and a stability theory based solution [22] to the equation error formulation of ARMA process identification prove identical this does not suggest the equivalence of gradient based [23] and stability theory based [24] solutions to the output error identification problem. At least currently, they are seen as special cases of a general, possible solution [25]. These two forms of adaptive parameter estimation underly the various provable adaptive control schemes suggesting differences for ROAC. Furthermore the distinction of equivalent convergence points versus convergence paths must be made. For full-order adaptive identification the asymptotic results may be similar though the transient behaviors are quite distinct. This transient disparity is amplified in reduced-order identification where the convergence points also become distinctly different [26]. Therefore, it would appear that the parameter estimation formulation and the explicit or implicit adaptive control strategy both lead to different ROAC behaviors. The fourth objective is to test the equivalence (or disparity) of indirect and direct adaptive controllers based on gradient and stability theory formulated equation and output error parameter estimators.

Achievement of the preceding objectives may provide suggestions for improving the presently available full-order adaptive controllers for ROAC use. Fulfillment of this hope is the ultimate objective of this report.

## III. TEST EXAMPLES

Six test examples will be used to meet the objectives stated in the preceding section. They represent the following six categories:

(i) Open-loop and closed-loop (in near satisfaction of desired model-following objective) system singularities are separable on a time basis due to significant damping differences.

(ii) Open-loop system singularities are time separable but closed-loop singularities are only marginally separable.

(iii) Neither open-loop nor closed-loop singularities are time separable but a near fit to desired behavior exists due to a lightly damped near-cancellation.

(iv) Neither open-loop nor closed-loop singularities are time separable and no reduced-order controller gain closely satisfies the desired control objective.

(v) Two nearly oscillatory open-loop modes of distinct frequency, which are only moderately shifted in the closed-loop, are preceded by an infinite bandwidth actuator.

(vi) Two nearly oscillatory open-loop modes of the same distinct frequencies as in (v) are preceded by a low pass actuator providing at least -12 db (75%) attenuation at the resonant frequency of the second mode relative to the resonant frequency of the first.

Categories (i)-(iv) will test points (i) and (iii) of the preceding section. Categories (v) and (vi) will test point (ii). All of the categories will test point (iv).

Consider, for the first four categories, controlling a stable second

order plant with transfer function

$$\frac{Y(z)}{U(z)} = \frac{b}{z-a_1} + \frac{\varepsilon}{z-a_2} = \frac{(b+\varepsilon)z - (ba_2+\varepsilon a_1)}{(z-a_1)(z-a_2)} \tag{3-1}$$

and therefore difference equation description

$$\begin{aligned} y(k) &= (a_1+a_2)y(k-1) - (a_1a_2)y(k-2) \\ &\quad + (b+\varepsilon)u(k-1) - (ba_2+\varepsilon a_1)u(k-2) \end{aligned} \tag{3-2}$$

and D.C. gain

$$\left.\frac{Y(z)}{U(z)}\right|_{z=1} = \frac{b}{1-a_1} + \frac{\varepsilon}{1-a_2} \tag{3-3}$$

under the assumption of a first order model for (3-1)with the objective of following a first order model

$$s(k) = cr(k-1) + ds(k-1) \; . \tag{3-4}$$

Using the projection technique of (1-6)-(1-7), (3-1) can be rewritten as

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} a_1 & 0 \\ 0 & a_2 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} b \\ \varepsilon \end{bmatrix} u(k) \tag{3-5}$$

$$y(k) = [1 \quad 1] \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} , \tag{3-6}$$

where, from (1-6)-(1-7), $A_N = a_1$, $A_R = a_2$, $A_{NR} = A_{RN} = 0$, $b_Nc_N = b$, and $b_Rc_R = \varepsilon$. Note that any proportion of $\varepsilon$ could be distributed to $b_R$ and $c_R$. Therefore increased $\varepsilon$ corresponds to increased spillover. Clearly when $\varepsilon = b_Rc_R = 0$, a first order model of (3-1) would be exact. In such a case, as noted earlier, any stable adaptive control scheme would be

successful. Also, equally apparent is the degradation of the reduced-order control system, adaptive or not, as $\varepsilon$ becomes nonzero and the residual mode $x_2$ contributes significantly to y.

If $\varepsilon$ were zero, then control of (3-1) via

$$u(k-1) = gr(k-1) + fy(k-1) \tag{3-7}$$

where

$$g = \frac{c}{b} \tag{3-8}$$

$$f = \frac{d-a_1}{b}, \tag{3-9}$$

would convert the plant output to

$$y(k) = a_1 y(k-1) + bu(k-1) = cr(k-1) + dy(k-1) . \tag{3-10}$$

Therefore the model-following error

$$s(k) - y(k) = d[s(k-1) - y(k-1)] \tag{3-11}$$

would decay to zero if (3-4) were stable. If $\varepsilon$ were not zero, use of (3-7) would convert (3-1) to

$$\frac{y(z)}{R(z)} = g\left\{\frac{(b+\varepsilon)z - (ba_2+\varepsilon a_1)}{(z-a_1)(z-a_2) - f[(b+\varepsilon)z - (ba_2+\varepsilon a_1)]}\right\}$$

$$= \frac{g(b+\varepsilon)(z-h)}{(z-p_1)(z-p_2)}, \tag{3-12}$$

the roots of the characteristic equation of which, i.e. $p_1$ and $p_2$, could be determined via a root locus. Note that f has a limited effect on the poles of (3-12) while g can be chosen to select the D.C. gain.

Alternatively the control of (3-1) (or (3-2)) could be chosen for (3-5)-(3-6) as

$$u(k) = gr(k) + fx_1(k) \;, \tag{3-13}$$

if the reduced-order model state is assumed available. This assumption is equivalent to assuming the possibility of measuring $y_s(k)$ in (1-4) in the first reduced-order model strategy in the introduction. The control of (3-13) can be viewed as partial state feedback (impractically requiring state availability)

$$u(k) = gr(k) + [f \quad 0] \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix}. \tag{3-14}$$

From (3-5), (3-6), and (3-14), (3-13) converts (3-1) to

$$\frac{Y(z)}{R(z)} = g \; [1 \quad 1] \left\{ zI - \begin{bmatrix} a_1 & 0 \\ 0 & a_2 \end{bmatrix} - \begin{bmatrix} b \\ \varepsilon \end{bmatrix} [f \quad 0] \right\}^{-1} \begin{bmatrix} b \\ \varepsilon \end{bmatrix}$$

$$= g \left\{ \frac{(b+\varepsilon)z - (ba_2+\varepsilon a_1)}{(z-a_1-bf)(z-a_2)} \right\} = \frac{g(b+\varepsilon)(z-h)}{(z-q_1)(z-q_2)} \;. \tag{3-15}$$

Note that relative to (3-12) only one pole is arbitrarily shifted, since $q_2 = a_2$, rather than both being shifted under the root locus constraint. Note that in either case, (3-12) or (3-15), $\varepsilon$ can be chosen in constructing the test examples for a particular open-loop (and closed-loop, due to pole-shifting only by the controllers) zero location h

$$h = \frac{ba_2+\varepsilon a_1}{b+\varepsilon} \Longrightarrow \varepsilon = \frac{b(a_2-h)}{h-a_1} \;. \tag{3-16}$$

Example 1: $a_1 = 0.95$, $b = 0.065$, $a_2 = 0.2$, $\varepsilon = 0.01$, $h = 0.3$, $c = 0.4$, $d = 0.8$

Recall that from (3-1) $a_1$ and $a_2$ are the open-loop plant poles, from (3-4) d is the desired closed-loop pole, from (3-12) and (3-15) $p_1$, $p_2$ and $q_1$,

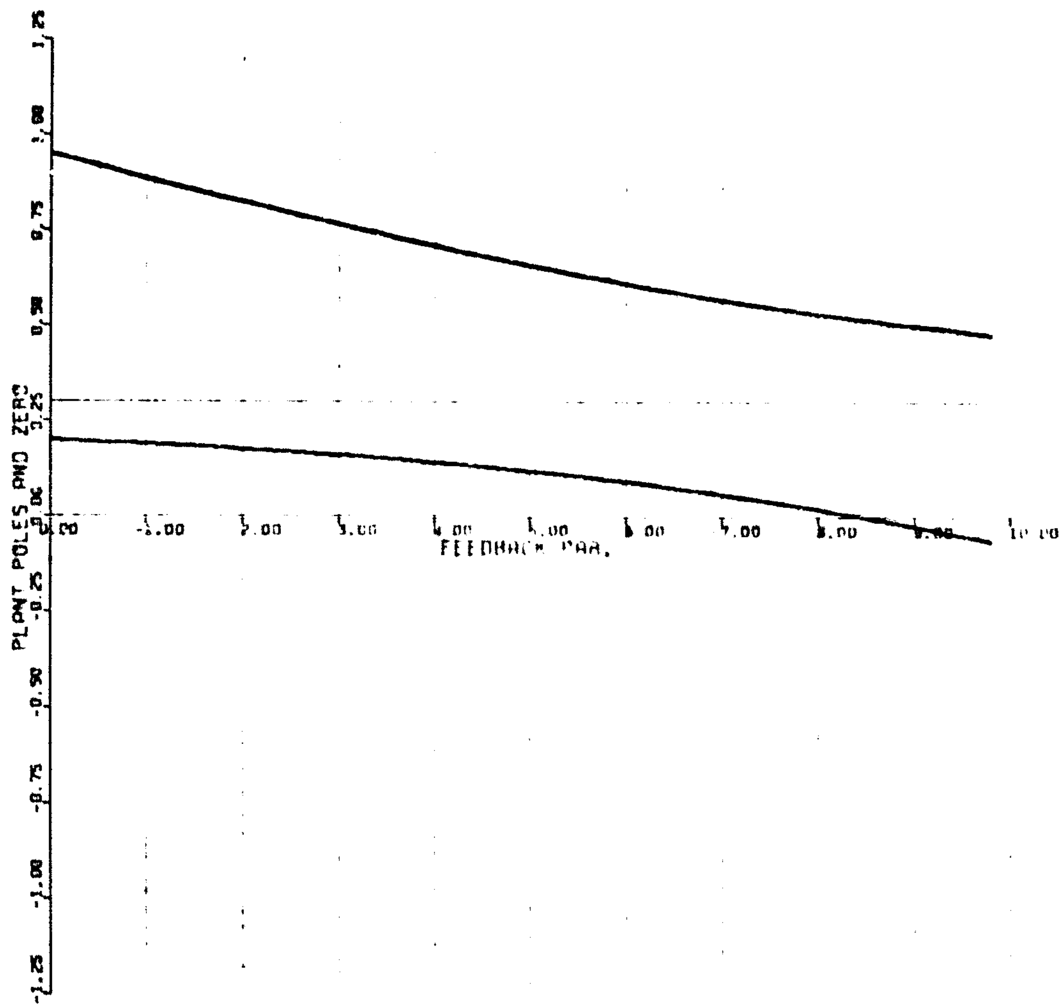
$q_2$ are the closed-loop system poles due to output and partial state feedback, respectively, and from (3-16) h is the open- and closed-loop system zero. The closed-loop migration due to various f in (3-7) and (3-13) appear in Fig. 3-1 and 3-2, respectively. Note that, since $a_1^5 \approx 0.77 > h > a_2$ and using (3-7) with $f = -2.4 \Rightarrow p_1^5 = (0.8)^5 \approx 0.33 > h > p_2 = 0.17$ and using (3-13) with $f \approx -2.3 \Rightarrow q_1^5 \doteq (0.8)^5 \approx 0.33 > h > q_2 = 0.2$, this example falls in category (i). Note the near equivalence of the f's from (3-7) or (3-13) for the same objective. (Refer to Appendix A for the program listings and tabular output of these and the following root locus plots.)

<u>Example 2</u>: $a_1 = 0.9$, $b = 0.1$, $a_2 = 0.1$, $\varepsilon = -0.01$, $h = 0.011$, $c = 0.7$,
$d = 0.65$

The closed-loop root migration due to various f in (3-7) and (3-13) appear in Fig. 3-3 and 3-4, respectively. Note that $a_1^5 \approx 0.59 > a_2 > h$ and $d^5 \approx 0.12 > a_2 > h$, i.e. the dominant open-loop pole and its desired location both dominate the other open-loop singularities, but for $f \doteq -2.4$ in (3-7) the second closed-loop pole $p_2 \approx 0.13$ is crossing the dominance threshold with respect to $p_1 = 0.65$ as shown in Fig. 3-3, thereby putting this example in category (ii). To retain the open-loop dominance the partial state feedback of (3-14) in Fig. 3-4 must be used.

<u>Example 3</u>: $a_1 = 0.9$, $b = 0.08$, $a_2 = 0.8$, $\varepsilon = 0.02$, $h = 0.82$, $c = 1$,
$d = 0.5$

The closed-loop root migration due to various output feedback f in (3-7) is plotted in Fig. 3-5. Figure 3-6 shows the closed-loop roots for various partial state feedback gains in (3-13). Reversing the designation of the




—POLES

—ZERO

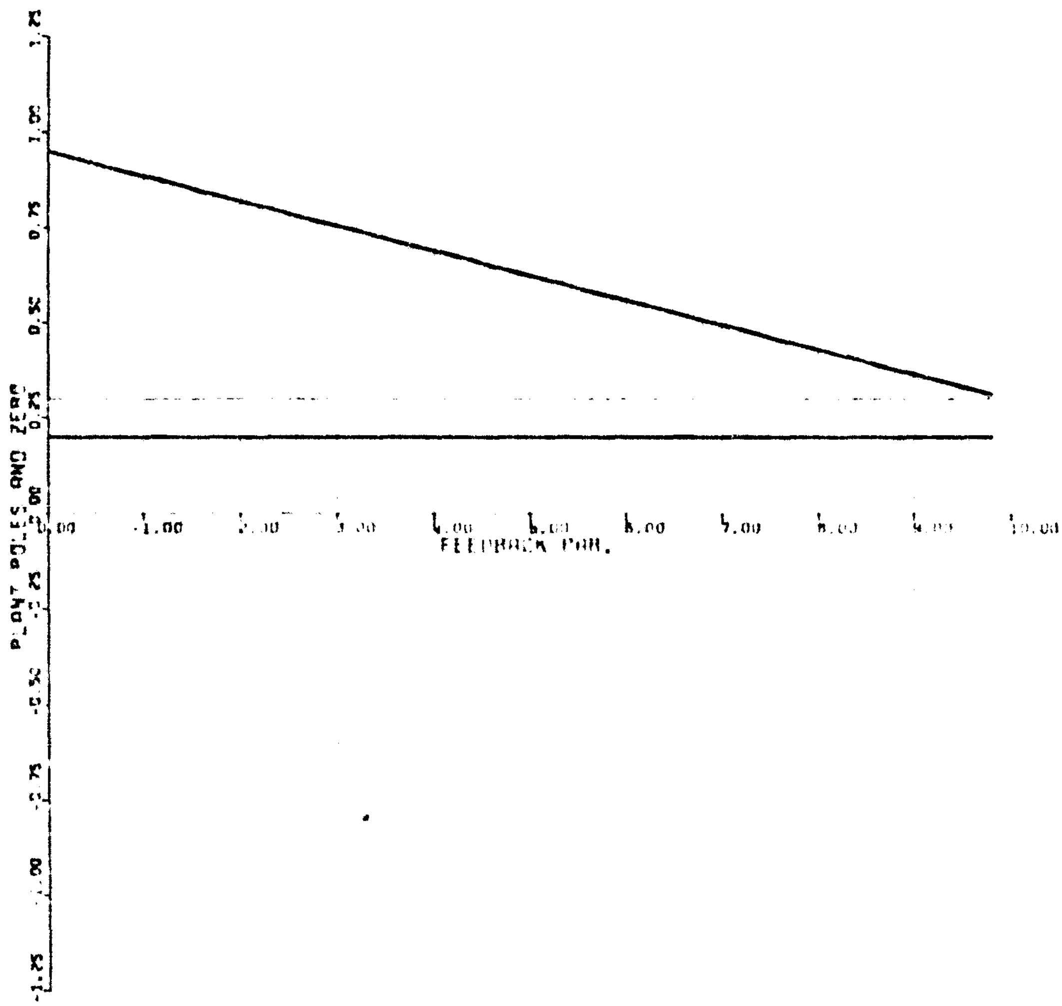
PLANT: Y(K)=(A1+A2)Y(K-1)-(A1*A2)Y(K-2)+(B+E)*U(K-1)-(A1*E+B*A2)*U(K-2)

CONTROL: U(K)=F1*R(K)+F2*Y(K)

RESULTING CHAR. EQN.: Z**2 Z*(A1+A2+F*(B+E))+A1*A2+F*(B*A2+E*A1)

PARAMETERS: A1-0.95 A2=0.2 B=0.065 E 0.01

Fig. 3-1: Closed-loop Root Migration due to Output Feedback for Example 1.




__POLES

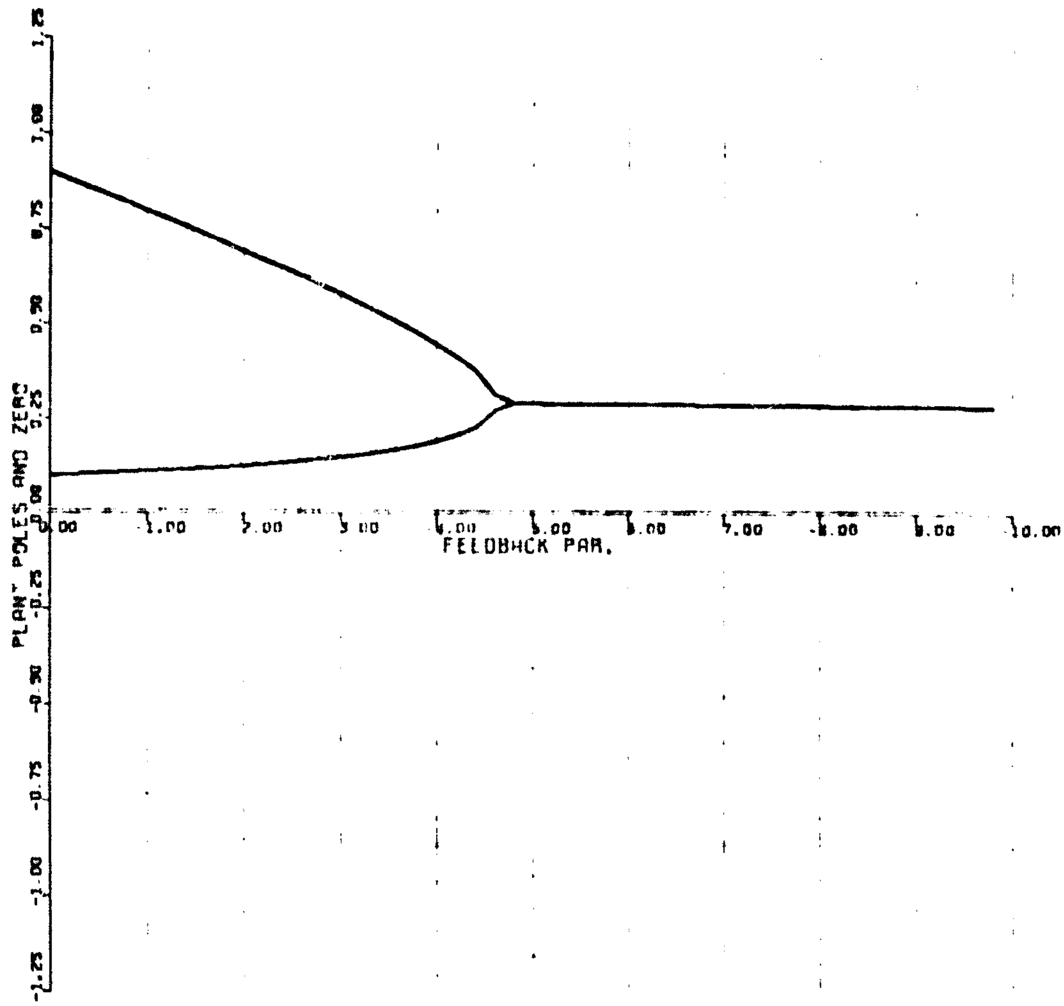
_.ZERO

PLANT: Y(K)=(A1+A2)*Y(K-1)-(A1*A2)*Y(K-2)+(B+E)*U(K-1)-(A1*E+B*A2)*U(K-2)

RESULTING CHAR. EQN.: Z**2-Z*(A1+A2+E*B)+A1*A2+B*A2*E

PARAMETERS: A1=0.95 A2=0.2 B=0.005 E=0.01

Fig. 3-2: Closed-loop Root Migration due to Partial State Feedback for Example 1.




___POLES (Merged locus plots magnitude of complex conjugates)

___ZERO

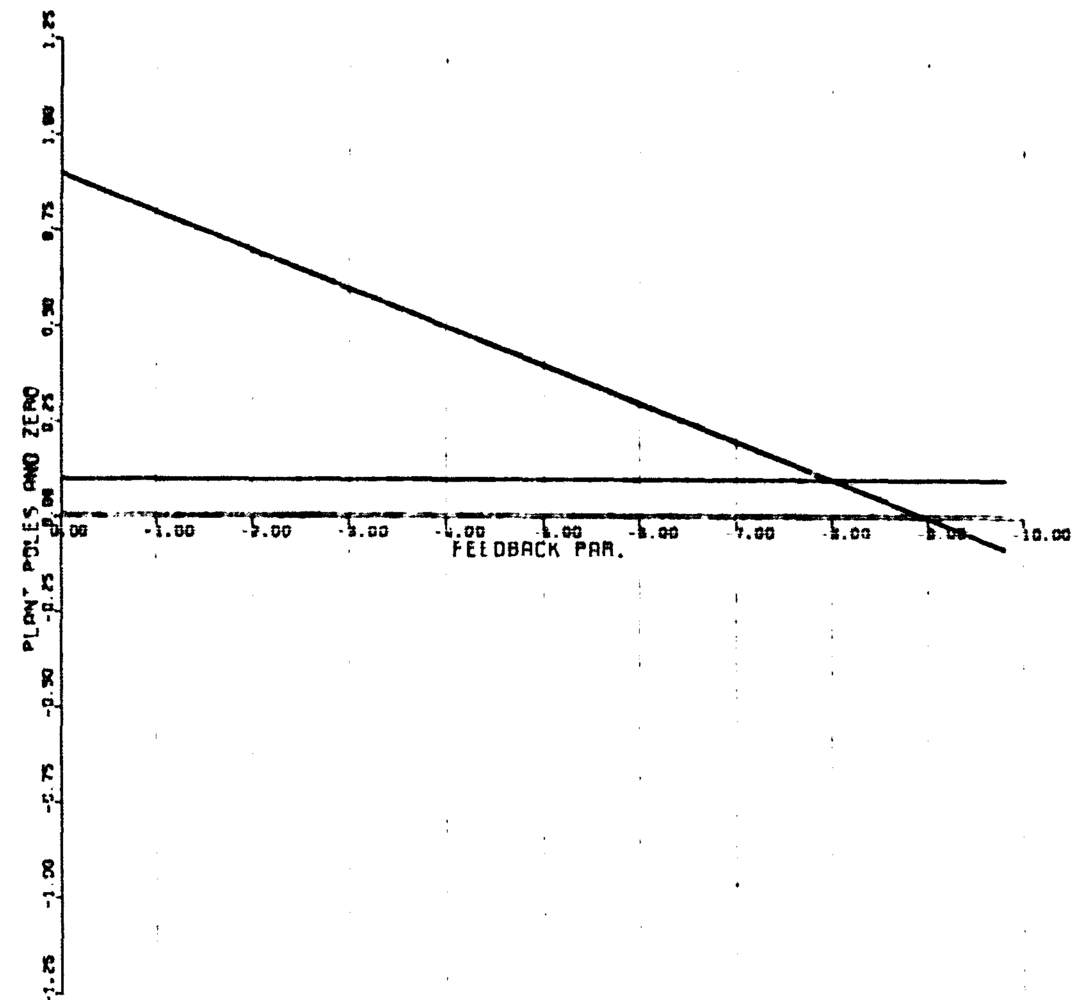
PLANT: Y(K)=(A1+A2)Y(K-1)-(A1*A2)Y(K-2)+(B+E)*U(K-1)-(A1*E+B*A2)*U(K-2)

CONTROL: U(K)-F1*R(K)+F2*Y(K)

RESULTING CHAR. EQN.: Z**2-Z*(A1+A2+F*(B+E))+A1*A2+F*(B*A2+E*A1)

PARAMETERS: A1=0.9 A2=0.1 B=0.1 E=-0.01

Fig. 3-3: Closed-loop Root Migration due to Output Feedback for Example 2.




__POLES

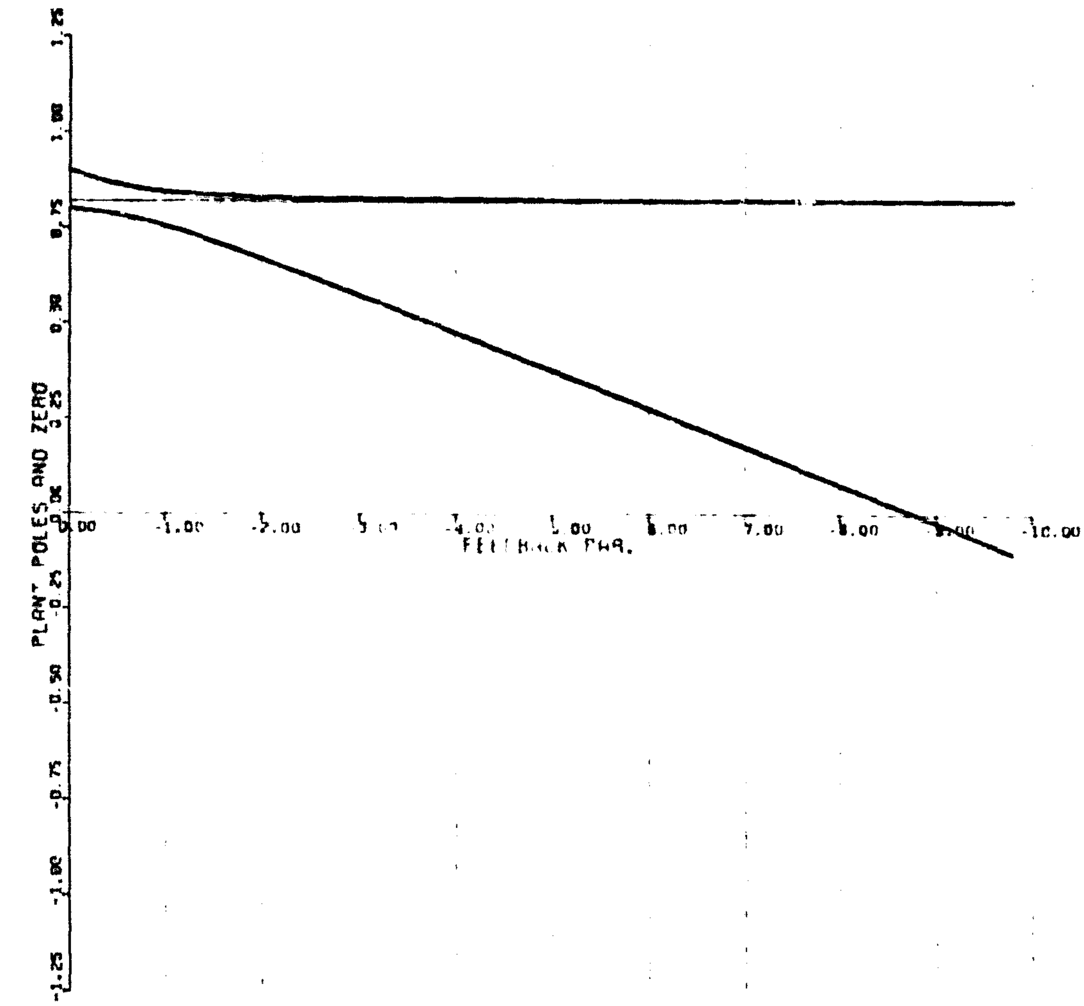
__ZERO

PLANT: Y(K)=(A1+A2)Y(K-1)-(A1*A2)Y(K-2)+(B+E)*U(K-1)-(A1*E+B*A2)*U(K-2)

RESULTING CHAR. EQN.: Z**2-Z*(A1+A2+F*B)+A1*A2+B*A2*F

PARAMETERS: A1=0.9    A2=0.1    B=0.1    E=-0.01

Fig. 3-4: Closed-loop Root Migration due to Partial State Feedback for Example 2.




__POLES

__ZERO

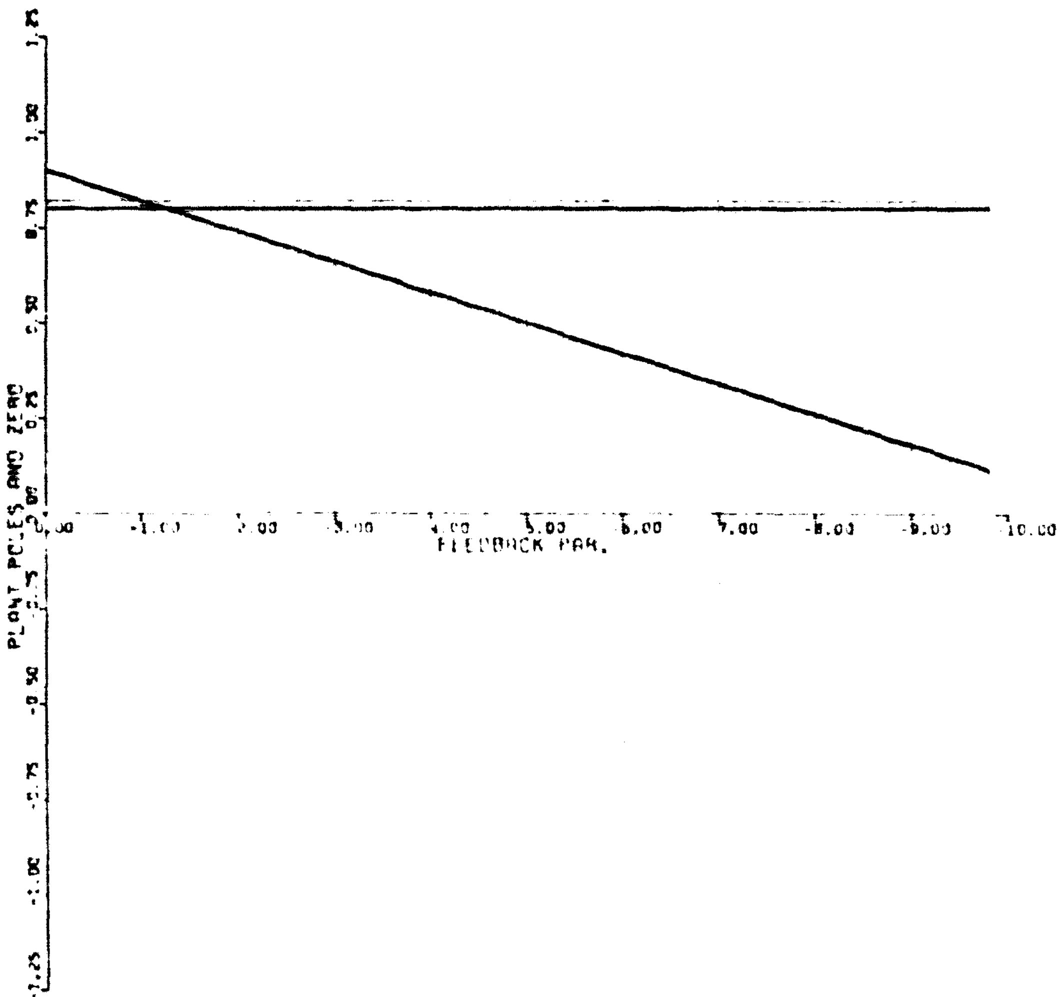
PLANT: Y(K) = (A1+A2)Y(K-1) - (A1*A2)Y(K-2) + (B+E)*U(K-1) - (A1*E+B*A2)*U(K-2)

CONTROL: U(K)=F1*R(K)+F2*Y(K)

RESULTING CHAR. EQN.: Z**2-Z*(A1+A2+F*(B+E))+A1*A2+F*(B*A2+E*A1)

PARAMETERS: A1=0.9 A2=0.8 B=0.08 E=0.02

Fig. 3-5: Closed-loop Root Migration due to Output Feedback for Example 3.




__POLES

_ _ ZERO

PLANT: Y(K)=(A1+A2)Y(K-1)-(A1*A2)Y(K-2)+(B+E)*U(K-1)-(A1*E+B*A2)*U(K-2)

RESULTING CHAR. EQN.: Z**2-Z*(A1+A2+F*B)+A1*A2+B*A2*F

PARAMETERS: A1=0.9 A2=0.8 B=0.[illegible] E=0.02

Fig. 3-6: Closed-loop Root Migration due to Partial State Feedback for Example 3.

important mode for partial state feedback by either swapping the elements of the state feedback gain vector in (3-14) or swapping the designation of $a_1$, b and $a_2$, $\varepsilon$ results in the closed-loop roots shown in Fig. 3-7. Note in Fig. 3-5 for $f \cong -3.8$ the near-cancellation of the slower mode while the faster mode matches that desired. The degree of closeness of this near cancellation is not matched in either Fig. 3-6 or 3-7. Note that in Fig. 3-6 if f is chosen as -5 to cause $q_1 = 0.5$, use of the same f in Fig. 3-5 would result in overcompensation. Neither in the open-loop plant nor for any f placing one pole near the desired location of d = 0.5 are the singularities separable as in category (iii).

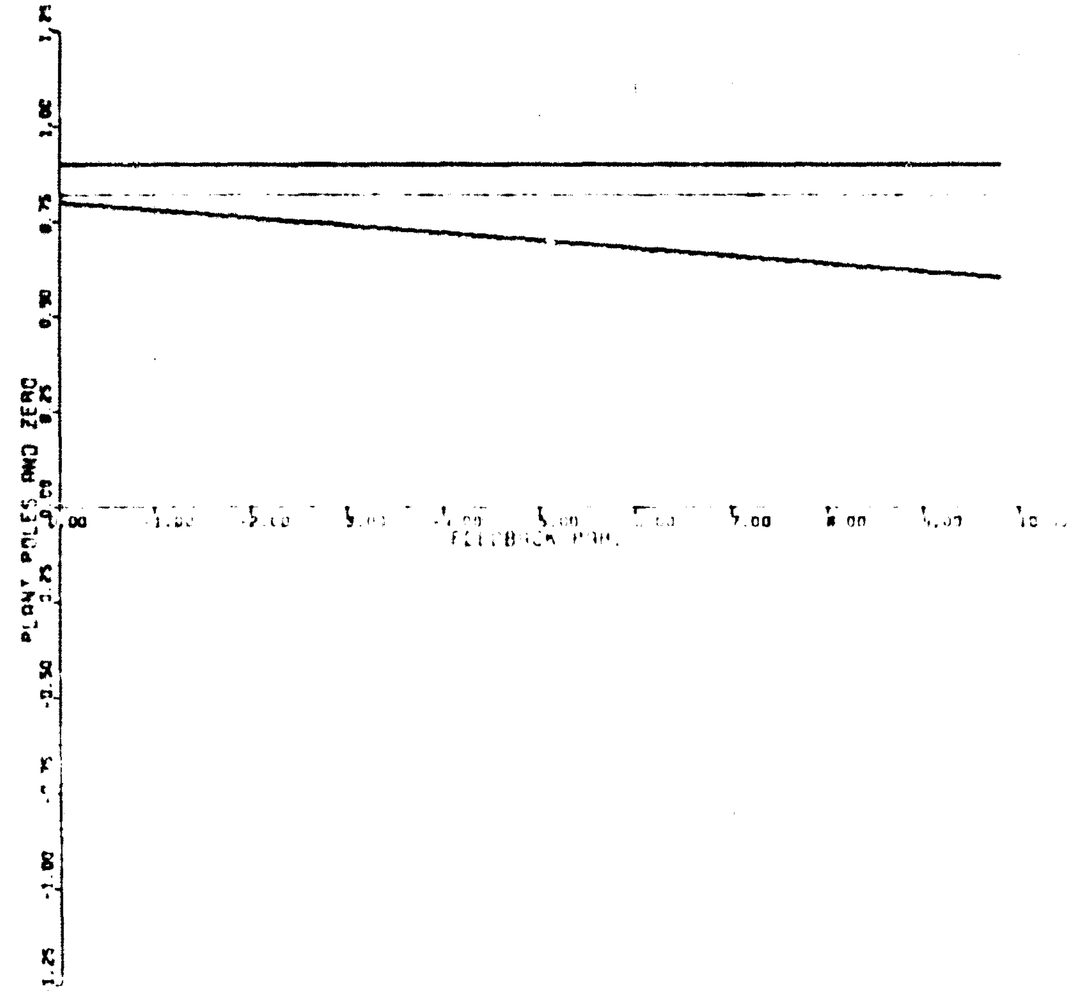
Example 4: $a_1 = 0.9$, $b = 0.5$, $a_2 = 0.7$, $\varepsilon = -0.3$, $h = 0.4$, $c = 0.8$, $d = 0.6$

The pole migration due to the output feedback f from (3-7) displays in Fig. 3-8 the oscillatory character of the second-order poles in the region of desired pole radius. Even partial state feedback proves unsuccessful as shown in Fig. 3-9 due to the retention of one of the poles at its nearer unity value than the desired pole. Therefore, this example falls into category (iv).

Consider, for the last two example categories, digitally controlling a stable fourth-order discrete equivalent of a continuous plant composed of two quadratic, all-pole, modes

$$\frac{Y(z)}{U(z)} = \frac{\beta_{11}z + \beta_{12}}{z^2 - \alpha_{11}z - \alpha_{12}} + \frac{\beta_{21}z + \beta_{22}}{z^2 - \alpha_{21}z - \alpha_{22}} , \tag{3-17}$$

where the $\alpha$ and $\beta$ are derived from the zero-order-hold equivalent




—POLES

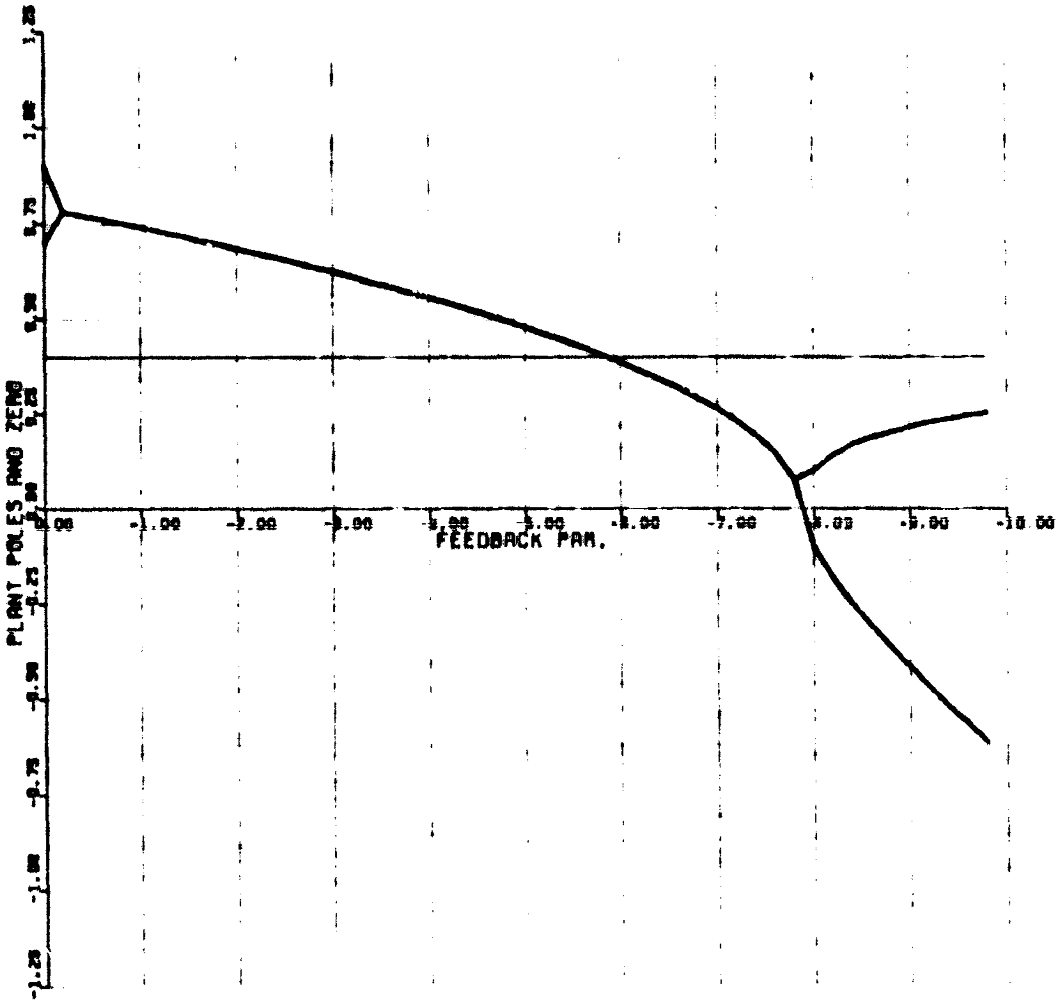
— ZERO

PLANT: [illegible]

RESULTING CHAR. EQN.: Z**2 Z*(A1+A2+F*E)+A1*A2+A1*E*F

PARAMETERS: A1=0.9 A2 [illegible] 0.8 B=0.08 E=0.02

Fig. 3-7: Closed-loop Root Migration due to Partial State Feedback for Example 3.




—POLES (Merged locus plots magnitude of complex conjugates)

—ZERO

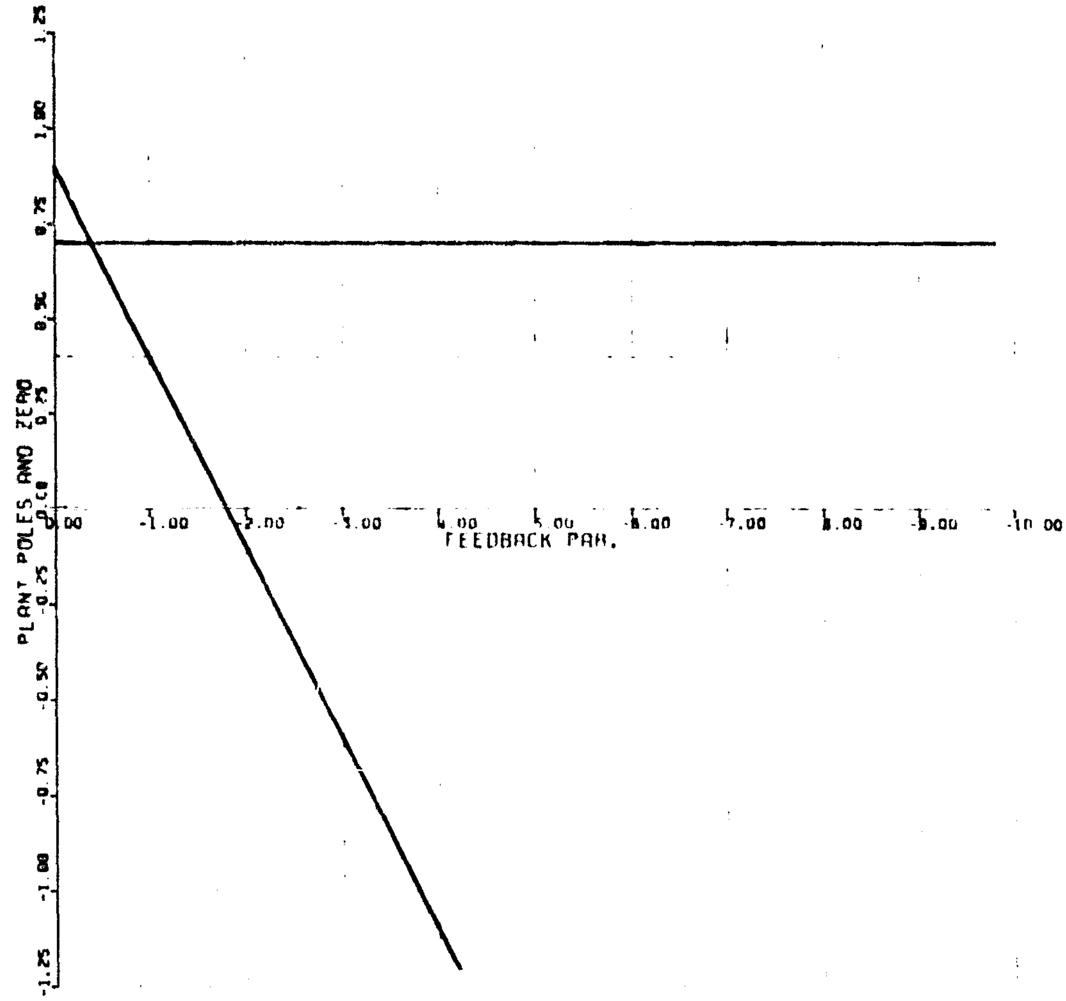
PLANT: Y(K)=(A1+A2)Y(K-1)-(A1*A2)Y(K-2)+(B+E)*U(K-1)-(A1*E+B*A2)*U(K-2)

CONTROL: U(K)=F1*R(K)+F2*Y(K)

RESULTING CHAR. EQN.: Z**2-Z*(A1+A2+F*(B+E))+A1*A2+F*(B*A2+E*A1)

PARAMETERS: A1=0.9 A2=0.7 B=0.5 E=-0.3

Fig. 3-8: Closed-loop Pole Migration due to Output Feedback for Example 4.






—— POLES

— — ZERO

PLANT: Y(K)=(A1+A2)Y(K-1)-(A1*A2)Y(K-2)+(B+E)*U(K-1)-(A1*E+B*A2)*U(K-2)

CONTROL: U(K)=F1*R(K)+F2*Y(K)

RESULTING CHAR. EQN.: Z**2-Z*(A1+A2+F*(B+E))+A1*A2+F*(B*A2+E*A1)

PARAMETERS: A1=0.9    A2=0.7    B=0.5    E=-0.3

Fig. 3-9: Closed-loop Pole Migration due to Partial State Feedback for Example 4.

$$\frac{\beta_{i1}z + \beta_{i2}}{z^2 - \alpha_{i1}z - \alpha_{i2}} = (1-z^{-1})Z\left[\left(\frac{1}{s}\right)\left\{\frac{\lambda_i}{s^2 + 2\zeta_i\omega_i s + \omega_i^2}\right\}\right] \tag{3-18}$$

with [27]

$$\alpha_{i1} = 2\, e^{-\zeta_i\omega_i T} \cos\left(\omega_i T\sqrt{1 - \zeta_i^2}\right) \tag{3-19}$$

$$\alpha_{i2} = -e^{-2\zeta_i\omega_i T} \tag{3-20}$$

$$\beta_{i1} = \frac{\lambda_i}{\omega_i^2}\left\{1 - e^{-\zeta_i\omega_i T}\left[\cos(\omega_i T\sqrt{1-\zeta_i^2}) + (\zeta_i/\sqrt{1-\zeta_i^2}) \sin(\omega_i T\sqrt{1-\zeta_i^2})\right]\right\} \tag{3-21}$$

$$\beta_{i2} = \frac{\lambda_i}{\omega_i^2}\left\{e^{-\zeta_i\omega_i T}\left[e^{-\zeta_i\omega_i T} - \cos(\omega_i T\sqrt{1-\zeta_i^2}) + (\zeta_i/\sqrt{1-\zeta_i^2}) \sin(\omega_i T\sqrt{1-\zeta_i^2})\right]\right\} . \tag{3-22}$$

The reduced-order, model-following objective is to track the output of

$$\frac{S(z)}{R(z)} = \frac{\delta_1 z + \delta_2}{z^2 - \gamma_1 z - \gamma_2} , \tag{3-23}$$

where $\gamma_1$ and $\gamma_2$ can represent the discrete equivalent of desired s-plane pole locations in terms of $\zeta$ and $\omega$ and translated via conversions similar to (3-19) and (3-20). For $\beta_{21} = \beta_{22} = 0$ this objective can only be achieved if the numerator of (3-17) is cancelled and replaced by that of (3-23). Therefore assume that the numerator zero of (3-23) matches that of (3-17) with $\beta_{21} = \beta_{22} = 0$ and only the poles require shifting. This converts the objective model in (3-23) to

$$\frac{S(z)}{R(z)} = \frac{\delta(\beta_{11}z + \beta_{12})}{z^2 - \gamma_1 z - \gamma_2} . \tag{3-24}$$

The control effort

$$u(k) = \delta r(k) + \eta_1 u(k-1) + \eta_2 u(k-2) + \nu_1 y(k-1) + \nu_2 y(k-2) \, , \tag{3-25}$$

where [28]

$$\begin{bmatrix} \eta_1 \\ \eta_2 \\ \nu_1 \\ \nu_2 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 & 0 \\ \alpha_{11} & -1 & -\beta_{11} & 0 \\ \alpha_{12} & \alpha_{11} & -\beta_{12} & -\beta_{11} \\ 0 & \alpha_{12} & 0 & -\beta_{12} \end{bmatrix}^{-1} \begin{bmatrix} \alpha_{11}-\gamma_1 \\ \alpha_{12}-\gamma_2 \\ 0 \\ 0 \end{bmatrix} \tag{3-26}$$

converts (3-17), with $\beta_{21} = \beta_{22} = 0$ to (3-24). An algebraic, sequentially calculated solution of (3-26) avoiding the matrix inversion is [3][9]

$$\eta_1 = \gamma_1 - \alpha_{11} \tag{3-27}$$

$$\nu_1 = \frac{\left[\eta_1\alpha_{12} + (\eta_1\alpha_{11} - \alpha_{12} + \gamma_2)(\alpha_{11} - \alpha_{12}\beta_{11}/\beta_{12})\right]}{\left[\beta_{11}\alpha_{11} + \beta_{12} - \alpha_{12}\beta_{11}^2/\beta_{12}\right]} \tag{3-28}$$

$$\eta_2 = \alpha_{11}\eta_1 + \gamma_2 - \beta_{11}\nu_1 - \alpha_{12} \tag{3-29}$$

$$\nu_2 = \eta_2\alpha_{12}/\beta_{12} \, . \tag{3-30}$$

If $\beta_{21}$ or $\beta_{22} \neq 0$ and (3-25) is used on (3-17) then

$$\frac{Y(z)}{R(z)} = \delta\left\{\frac{\left[1/(1 - \eta_1 z^{-1} - \eta_2 z^{-2})\right]\left[Y(z)/U(z)\right]}{1 - \left[(\nu_1 z^{-1} + \nu_2 z^{-2})/(1 - \eta_1 z^{-1} - \eta_2 z^{-2})\right]\left[Y(z)/U(z)\right]}\right\}$$

$$= \frac{\delta z^2\left[(\beta_{11}z+\beta_{12})(z^2-\alpha_{21}z-\alpha_{22}) + \varepsilon(\beta_{21}z+\beta_{22})(z^2-\alpha_{11}z-\alpha_{12})\right]}{\left\{(z^2-\eta_1 z-\eta_2)(z^2-\alpha_{11}z-\alpha_{12})(z^2-\alpha_{21}z-\alpha_{22}) - (\nu_1 z+\nu_2)\left[(\beta_{11}z+\beta_{12})(z^2-\alpha_{21}z-\alpha_{22}) + \varepsilon(\beta_{21}z+\beta_{22})(z^2-\alpha_{11}z-\alpha_{12})\right]\right\}}$$

$$= \frac{b_5 z^5 + b_4 z^4 + b_3 z^3 + b_2 z^2 + b_1 z + b_0}{a_6 z^6 + a_5 z^5 + a_4 z^4 + a_3 z^3 + a_2 z^2 + a_1 z + a_0} \tag{3-31}$$

where

$$b_5 = \delta(\beta_{11} + \beta_{21}) \tag{3-32}$$

$$b_4 = \delta(-\beta_{11}\alpha_{21} + \beta_{12} - \beta_{21}\alpha_{11} + \beta_{22}) \tag{3-33}$$

$$b_3 = \delta(-\beta_{11}\alpha_{22} - \beta_{12}\alpha_{21} - \beta_{21}\alpha_{12} - \beta_{22}\alpha_{11}) \tag{3-34}$$

$$b_2 = \delta(-\beta_{12}\alpha_{22} - \beta_{22}\alpha_{12}) \tag{3-35}$$

$$b_1 = b_0 = 0 \tag{3-36}$$

$$a_6 = 1 \tag{3-37}$$

$$a_5 = -\alpha_{21} - \alpha_{11} - \eta_1 \tag{3-38}$$

$$a_4 = -\alpha_{22} + \alpha_{11}\alpha_{21} - \alpha_{12} + \eta_1\alpha_{21} + \eta_1\alpha_{11} - \eta_2 - \nu_1\beta_{11} - \nu_1\beta_{21} \tag{3-39}$$

$$\begin{aligned} a_3 = {} & \alpha_{11}\alpha_{22} + \alpha_{12}\alpha_{21} + \eta_1\alpha_{22} - \eta_1\alpha_{11}\alpha_{21} + \eta_1\alpha_{12} + \eta_2\alpha_{21} + \eta_2\alpha_{11} \\ & + \nu_1\beta_{11}\alpha_{21} - \nu_1\beta_{12} + \nu_1\beta_{21}\alpha_{11} - \nu_1\beta_{22} - \nu_2\beta_{11} - \nu_2\beta_{21} \end{aligned} \tag{3-40}$$

$$\begin{aligned} a_2 = {} & \alpha_{12}\alpha_{22} - \eta_1\alpha_{11}\alpha_{22} - \eta_1\alpha_{12}\alpha_{21} + \eta_2\alpha_{22} - \eta_2\alpha_{11}\alpha_{21} + \eta_2\alpha_{12} \\ & + \nu_1\beta_{11}\alpha_{22} + \nu_1\beta_{12}\alpha_{21} + \nu_1\beta_{21}\alpha_{12} + \nu_1\beta_{22}\alpha_{11} \\ & + \nu_2\beta_{11}\alpha_{21} - \nu_2\beta_{12} + \nu_2\beta_{21}\alpha_{11} - \nu_2\beta_{22} \end{aligned} \tag{3-41}$$

$$\begin{aligned} a_1 = {} & -\eta_1\alpha_{12}\alpha_{22} - \eta_2\alpha_{11}\alpha_{22} - \eta_2\alpha_{12}\alpha_{21} + \nu_1\beta_{12}\alpha_{22} + \nu_1\beta_{22}\alpha_{12} \\ & + \nu_2\beta_{11}\alpha_{22} + \nu_2\beta_{12}\alpha_{21} + \nu_2\beta_{21}\alpha_{12} + \nu_2\beta_{22}\alpha_{11} \end{aligned} \tag{3-42}$$

$$a_0 = -\eta_2\alpha_{12}\alpha_{22} + \nu_2\beta_{12}\alpha_{22} + \nu_2\beta_{22}\alpha_{12} \, . \tag{3-43}$$

Note that $\frac{Y(z)}{R(z)}$ has the same numerator as $\frac{Y(z)}{U(z)}$. (See Appendix B for the supporting algebra and simulated check for (3-31)-(3-43) for the following example.)

Example 5: $T = 0.5$, $\lambda_1 = 1$, $\lambda_2 = 1$, $\zeta_1 = 0.2$, $\omega_1 = 0.5$, $\zeta_2 = 0.02$, $\omega_2 = 5$, $\delta = 0.5$, $\gamma_1 = 1.687$, $\gamma_2 = -0.741$.

The plant in (3-17) resulting from this parameterization has the pole-zero pattern shown in Fig. 3-10. Note $\zeta_i\omega_i T = 0.05$, i.e. both modes have the same settling time and are therefore not separable on a time basis. However, they are clearly separable on a frequency basis. The $\zeta_i$, $\omega_i$, and T were chosen to avoid aliasing, retain frequency separability, and model the lightly damped situation of flexible spacecraft [3][10]. Further, note that the low frequency mode has a DC gain which is 100 times that of the high frequency mode. Attempting to increase the damping ratio by a factor of three to 0.6 for the low frequency mode (and therefore third the settling time with $\omega_1$ unchanged) leads to the stated objective, which from (3-26) or (3-27)-(3-30) yields the following controller parameterization for (3-25): $\delta = 0.5$, $\eta_1 = -0.159$, $\eta_2 = -0.0715$, $\nu_1 = -0.475$, and $\nu_2 = 0.556$. Factorization of (3-31), where $b_5 = 0.952$, $b_4 = 1.188$, $b_3 = 1.128$, $b_2 = 0.83$, $b_1 = b_0 = 0$, $a_6 = 1$, $a_5 = -0.164$, $a_4 = -0.891$, $a_3 = -0.466$, $a_2 = 0.676$, $a_1 = 0.0625$, and $a_0 = -0.0338$ from (3-32)-(3-43), yields the pole-zero constellation in Fig. 3-11. (See Appendix C for the factorization and plotting routines used.) Note the closed-loop retention of frequency separability. Simply changing $\lambda_2$ to 10, thereby reducing the DC gain ratio from 100:1 to 10:1, leaves the plant poles of $0.923 \pm j0.231$

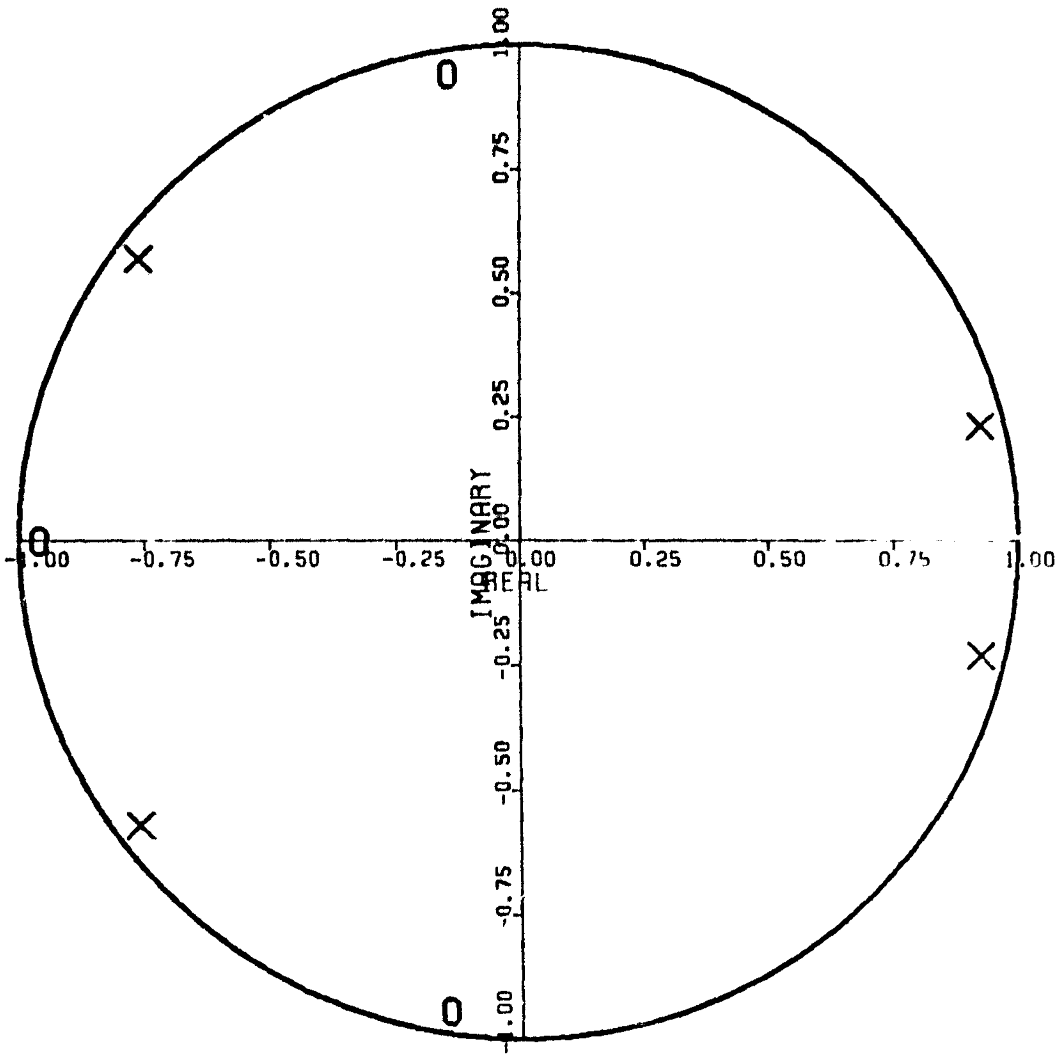



Figure 3-10: Plant Pole-zero Plot of Example 5

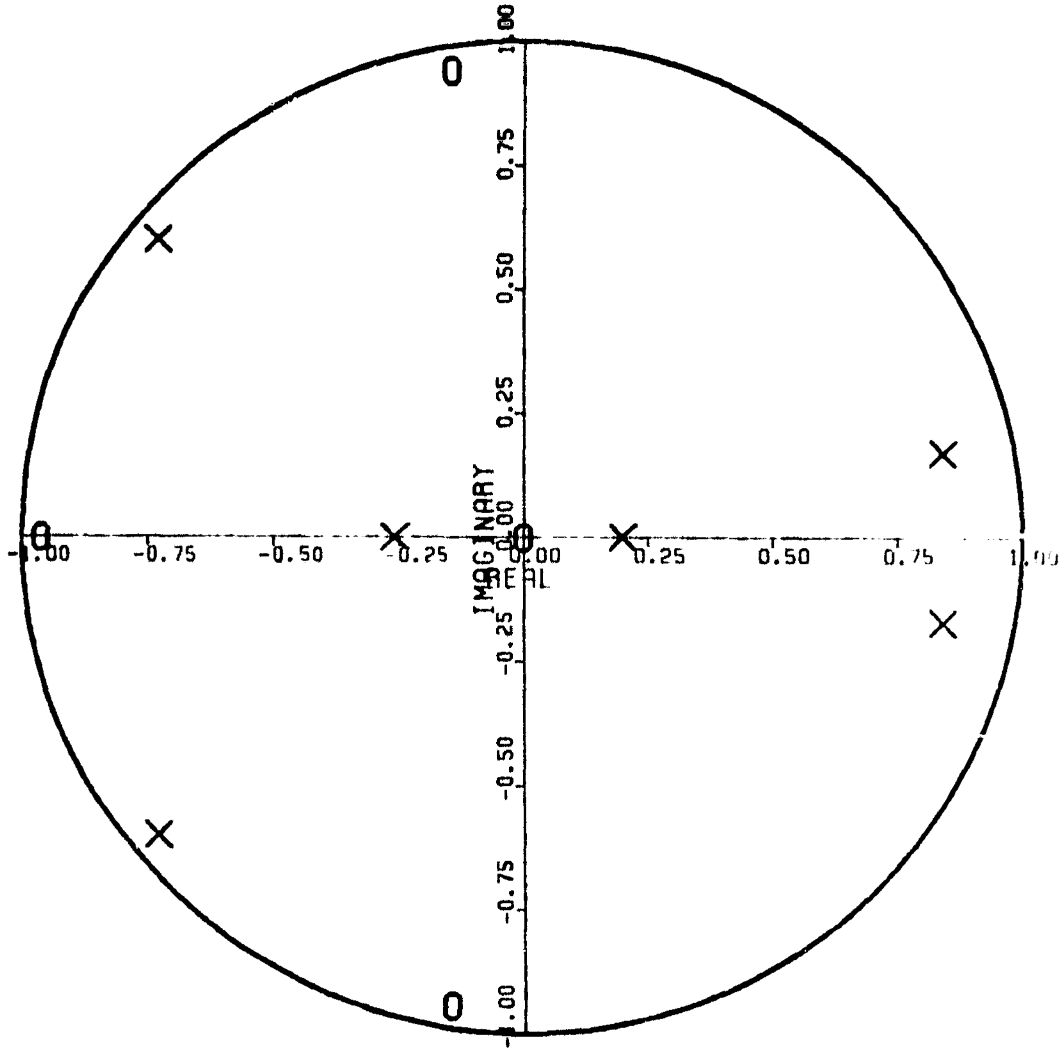



Figure 3-11: Reduced-Order Controlled Plant Pole-zero Plot for Example 5

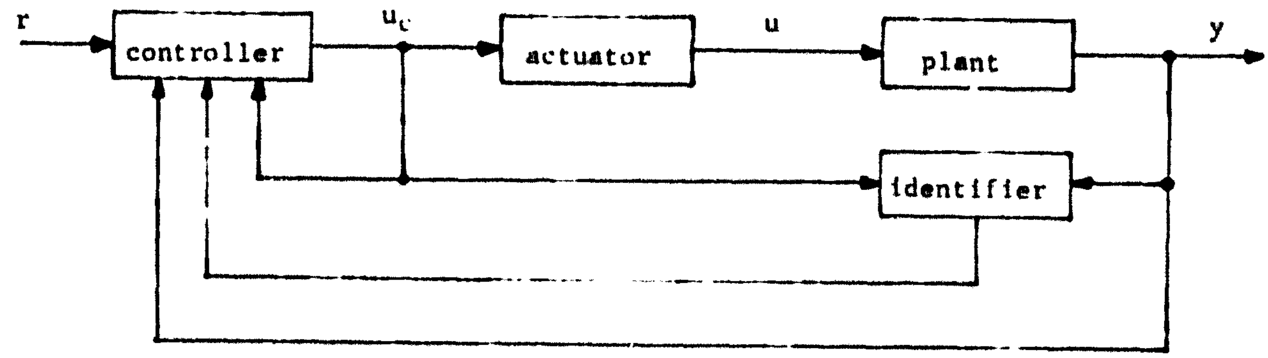
and $-0.762 \pm j0.570$ unaltered but changes the zeros from $-0.960$ and $-0.144 \pm j0.942$ to $-0.959$ and $0.675 \pm j0.671$. Since $\lambda_2$ does not affect the calculation of $\alpha_{1j}$ or $\beta_{1j}$ in (3-19)-(3-22), the $\eta_i$ and $\nu_i$ selected in (3-27)-(3-30) will be the same. However, this control law results in an unstable system. (See Appendix C.) Therefore observation spillover can be seen to destabilize an otherwise stable system if neglected in reduced-order controller design.

The addition of a low-pass actuator in controlling (3-17) would occur in the feedback loop after the dynamic output feedback element of (3-23) (with $\delta = 1$), if the reference signal were considered an unmeasurable disturbance. However the scaling of r in (3-25) suggests a command signal designation of r. Therefore the actuator would precede the plant in the forward path. A remaining question centers on the availability of the actuator output for control and identification. The three meaningful possibilities are diagrammed in Fig. 3-12a-c, where the identifier feeds back the parameterization of the plant for adaptive controller parameterization. Retention of the controller dynamics from (3-25) suggests usage of a or c. The frequency limited identification concept of "fact" (ii) in the preceding section implicit in example category (vi) prompts usage of b or c. Therefore c, if physically feasible, appears to be the prime candidate. However a is the most reasonable if the actuator models a physical limitation rather than a control logic device.

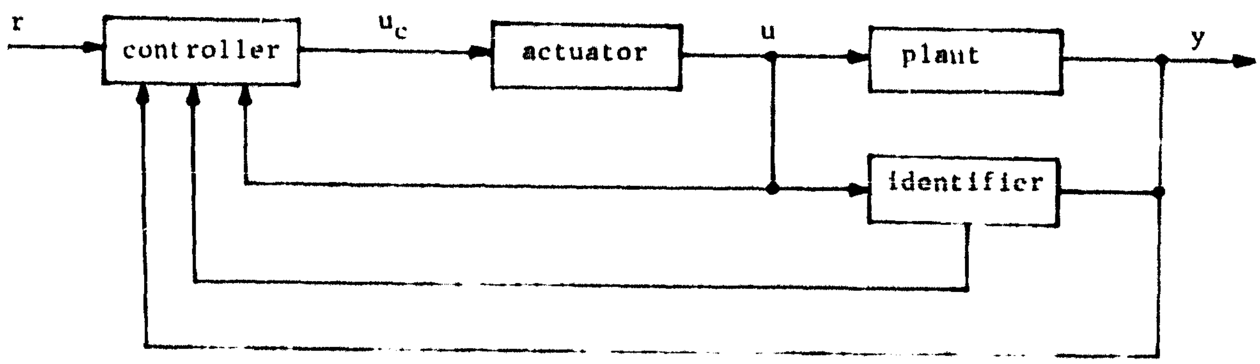
For the simple low-pass actuator model in [15] the zero-order-hold equivalent is [29]

$$H(z) = (1-z^{-1})\, Z\left\{\left(\frac{1}{s}\right)\left(\frac{\sigma}{s+\sigma}\right)\right\} = \frac{1-e^{-\sigma T}}{z-e^{-\sigma T}} \quad . \tag{3-44}$$

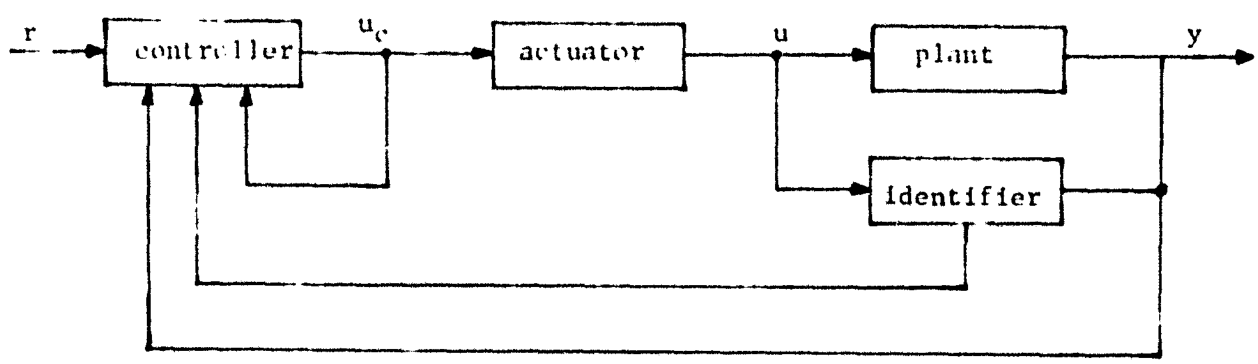
Therefore




(a) Actuator Incorporated in Plant




(b) Actuator Incorporated in Controller




(c) Actuator as Separable Element

Figure 3-12: Actuator Inclusion Possibilities

$$u(k+1) = e^{-\sigma T} u(k) + (1 - e^{-\sigma T}) u_c(k) , \tag{3-45}$$

where from (3-25) for Fig. 3-12a and c

$$u_c(k) = \delta r(k) + \eta_1 u_c(k-1) + \eta_2 u_c(k-2) + \nu_1 y(k-1) + \nu_2 y(k-2) \tag{3-46}$$

and for Fig. 3-12b

$$u_c(k) = \delta r(k) + \eta_1 u(k-1) + \eta_2 u(k-2) + \nu_1 y(k-1) + \nu_2 y(k-2) . \tag{3-47}$$

Applying (3-45)-(3-46) to (3-17) with the controller parameterization arising from e.g. (3-27)-(3-30), i.e. the reduced-order controller ignoring the actuator inclusion, yields the following overall transfer function (as shown in Appendix D)

$$\frac{Y(z)}{R(z)} = \frac{q_5 z^5 + q_4 z^4 + q_3 z^3 + q_2 z^2 + q_1 z + q_0}{z^7 + p_6 z^6 + p_5 z^5 + p_4 z^4 + p_3 z^3 + p_2 z^2 + p_1 z + p_0} \tag{3-48}$$

where

$$q_5 = \delta(1 - e^{-\sigma T})(\beta_{11} + \beta_{21}) \tag{3-49}$$

$$q_4 = \delta(1 - e^{-\sigma T})(-\beta_{11}\alpha_{21} + \beta_{12} - \beta_{21}\alpha_{11} + \beta_{22}) \tag{3-50}$$

$$q_3 = \delta(1 - e^{-\sigma T})(-\beta_{11}\alpha_{22} - \beta_{12}\alpha_{21} - \beta_{21}\alpha_{12} - \beta_{22}\alpha_{11}) \tag{3-51}$$

$$q_2 = \delta(1 - e^{-\sigma T})(-\beta_{12}\alpha_{22} - \beta_{22}\alpha_{12}) \tag{3-52}$$

$$q_1 = q_0 = 0 \tag{3-53}$$

$$p_6 = -e^{-\sigma T} - \alpha_{21} - \alpha_{11} - \eta_1 \tag{3-54}$$

$$p_5 = e^{-\sigma T}(\alpha_{21} + \alpha_{11} + \eta_1) - \alpha_{22} + \alpha_{11}\alpha_{21} - \alpha_{12} + \eta_1\alpha_{21}$$
$$+ \eta_1\alpha_{11} - \eta_2 \qquad (3\text{-}55)$$

$$p_4 = -e^{-\sigma T}(-\alpha_{22} + \alpha_{11}\alpha_{21} - \alpha_{12} + \eta_1\alpha_{21} + \eta_1\alpha_{11} - \eta_2) + \alpha_{11}\alpha_{22}$$
$$+ \alpha_{12}\alpha_{21} + \eta_1\alpha_{22} - \eta_1\alpha_{11}\alpha_{21} + \eta_1\alpha_{12} + \eta_2\alpha_{21} + \eta_2\alpha_{11}$$
$$- \nu_1(\beta_{11} + \beta_{21})(1 - e^{-\sigma T}) \qquad (3\text{-}56)$$

$$p_3 = -e^{-\sigma T}(\alpha_{11}\alpha_{22} + \alpha_{12}\alpha_{21} + \eta_1\alpha_{22} - \eta_1\alpha_{11}\alpha_{21} + \eta_1\alpha_{12} + \eta_2\alpha_{21} + \eta_2\alpha_{11})$$
$$+ \alpha_{12}\alpha_{22} - \eta_1\alpha_{11}\alpha_{22} - \eta_1\alpha_{12}\alpha_{21} + \eta_2\alpha_{22} - \eta_2\alpha_{11}\alpha_{21} + \eta_2\alpha_{12}$$
$$- \nu_1(-\beta_{11}\alpha_{21} + \beta_{22} + \beta_{12} - \beta_{21}\alpha_{11})(1 - e^{-\sigma T})$$
$$- \nu_2(\beta_{11} + \beta_{21})(1 - e^{-\sigma T}) \qquad (3\text{-}57)$$

$$p_2 = -e^{-\sigma T}(\alpha_{12}\alpha_{22} - \eta_1\alpha_{11}\alpha_{22} - \eta_1\alpha_{12}\alpha_{21} + \eta_2\alpha_{22} - \eta_2\alpha_{11}\alpha_{21} + \eta_2\alpha_{12})$$
$$- \eta_1\alpha_{12}\alpha_{22} - \eta_2\alpha_{11}\alpha_{22} - \eta_2\alpha_{12}\alpha_{21} + \nu_1(\beta_{11}\alpha_{22} + \beta_{12}\alpha_{21}$$
$$+ \beta_{21}\alpha_{12} + \beta_{22}\alpha_{11})(1 - e^{-\sigma T}) - \nu_2(-\beta_{11}\alpha_{21} + \beta_{22} + \beta_{12}$$
$$- \beta_{21}\alpha_{11})(1 - e^{-\sigma T}) \qquad (3\text{-}58)$$

$$p_1 = e^{-\sigma T}(\eta_1\alpha_{12}\alpha_{22} + \eta_2\alpha_{11}\alpha_{22} + \eta_2\alpha_{12}\alpha_{21}) - \eta_2\alpha_{12}\alpha_{22}$$
$$+ \nu_2(\beta_{11}\alpha_{22} + \beta_{12}\alpha_{21} + \beta_{21}\alpha_{12} + \beta_{22}\alpha_{11})(1 - e^{-\sigma T})$$
$$+ \nu_1(\beta_{12}\alpha_{22} + \beta_{22}\alpha_{12})(1 - e^{-\sigma T}) \qquad (3\text{-}59)$$

$$p_0 = e^{-\sigma T}\eta_2\alpha_{12}\alpha_{22} + \nu_2(1 - e^{-\sigma T})(\beta_{12}\alpha_{22} + \beta_{22}\alpha_{12}) \qquad (3\text{-}60)$$

Similarly the overall transfer function resulting from application of (3-45) and (3-47) to (3-17) is

$$\frac{Y(z)}{R(z)} = \frac{q_5 z^5 + q_4 z^4 + q_3 z^3 + q_2 z^2 + q_1 z + q_0}{z^7 + p_6 z^6 + p_5 z^5 + p_4 z^4 + p_3 z^3 + p_2 z^2 + p_1 z + p_0} \qquad (3\text{-}61)$$

where the $q_i$ are as in (3-49)-(3-53) and

$$p_6 = -\alpha_{21} - \alpha_{11} - e^{-\sigma T} \qquad (3\text{-}62)$$

$$p_5 = -\alpha_{22} + \alpha_{11}\alpha_{21} - \alpha_{12} + e^{-\sigma T}(\alpha_{21} + \alpha_{11}) - \eta_1(1 - e^{-\sigma T}) \qquad (3\text{-}63)$$

$$\begin{aligned} p_4 = {} & \alpha_{11}\alpha_{22} + \alpha_{12}\alpha_{21} - e^{-\sigma T}(-\alpha_{22} + \alpha_{11}\alpha_{21} - \alpha_{12}) \\ & + \eta_1(1 - e^{-\sigma T})(\alpha_{21} + \alpha_{11}) - \eta_2(1 - e^{-\sigma T}) - \nu_1(1 - e^{-\sigma T})(\beta_{11} + \beta_{21}) \end{aligned} \qquad (3\text{-}64)$$

$$\begin{aligned} p_3 = {} & \alpha_{12}\alpha_{22} - e^{-\sigma T}(\alpha_{11}\alpha_{22} + \alpha_{12}\alpha_{21}) - \eta_1(1 - e^{-\sigma T})(-\alpha_{22} + \alpha_{11}\alpha_{21} \\ & - \alpha_{12}) - \eta_2(1 - e^{-\sigma T})(-\alpha_{21} - \alpha_{11}) - \nu_1(1 - e^{-\sigma T})(-\beta_{11}\alpha_{21} \\ & + \beta_{22} + \beta_{12} - \beta_{21}\alpha_{11}) - \nu_2(1 - e^{-\sigma T})(\beta_{11} + \beta_{21}) \end{aligned} \qquad (3\text{-}65)$$

$$\begin{aligned} p_2 = {} & -e^{-\sigma T}\alpha_{12}\alpha_{22} - \eta_1(1 - e^{-\sigma T})(\alpha_{11}\alpha_{22} + \alpha_{12}\alpha_{21}) \\ & - \eta_2(1 - e^{-\sigma T})(-\alpha_{22} + \alpha_{11}\alpha_{21} - \alpha_{12}) + \nu_1(1 - e^{-\sigma T})(\beta_{11}\alpha_{22} \\ & + \beta_{12}\alpha_{21} + \beta_{21}\alpha_{12} + \beta_{22}\alpha_{11}) - \nu_2(1 - e^{-\sigma T})(-\beta_{11}\alpha_{21} \\ & + \beta_{22} + \beta_{12} - \beta_{21}\alpha_{11}) \end{aligned} \qquad (3\text{-}66)$$

$$\begin{aligned} p_1 = {} & -\eta_1(1 - e^{-\sigma T})\alpha_{12}\alpha_{22} - \eta_2(1 - e^{-\sigma T})(\alpha_{11}\alpha_{22} + \alpha_{12}\alpha_{21}) \\ & + \nu_2(1 - e^{-\sigma T})(\beta_{11}\alpha_{22} + \beta_{12}\alpha_{21} + \beta_{21}\alpha_{12} + \beta_{22}\alpha_{11}) \\ & + \nu_1(1 - e^{-\sigma T})(\beta_{12}\alpha_{22} + \beta_{22}\alpha_{12}) \end{aligned} \qquad (3\text{-}67)$$

$$p_0 = -\eta_2(1 - e^{-\sigma T})\alpha_{12}\alpha_{22} + \nu_2(1 - e^{-\sigma T})(\beta_{12}\alpha_{22} + \beta_{22}\alpha_{12}) \qquad (3\text{-}68)$$

<u>Example 6</u>: $T = 0.5,\ \lambda_1 = 1,\ \lambda_2 = 1,\ \zeta_1 = 0.2,\ \omega_1 = 0.5,\ \zeta_2 = 0.02,$

$\omega_2 = 5,\ \sigma = 1,\ \delta = 0.5,\ \gamma_1 = 1.687,\ \gamma_2 = -0.741$

Note that

$$H(s)\Big|_{s=0.5j} = \frac{\sigma}{0.5j + \sigma} = \frac{1}{1.12\angle 26.57^\circ} = 0.89\angle -26.57^\circ \qquad (3\text{-}69)$$

and

$$H(s)\Big|_{s=5j} = \frac{1}{5.1\angle 78.69^\circ} = 0.20\angle 78.69^\circ \ . \qquad (3\text{-}70)$$

Therefore, since [30, p. 387] $\omega_r = \omega_n\sqrt{1 - 2\zeta^2} \approx \omega_n$ for $\zeta \leq 0.2$, the actuator provides $20 \log\left(\frac{0.20}{0.89}\right) \approx -13$ db magnitude attenuation of the frequency response at the resonance of the high frequency mode versus the resonance of the low frequency mode in Example 5. Therefore Example 6 falls into category (vi). Note that $z = e^{-\sigma T} \approx 0.61 < e^{-5\zeta_1\omega_1 T} = e^{-5\zeta_2\omega_2 T} = (.95)^5 \approx 0.78$. However, the pole-shifting desired is to a radius of 0.861. Therefore the closed-loop transient effect of the actuator on the low frequency mode is expected to be appreciable but possibly tolerable. Despite the unity DC gain of H(z), its effect on the pole-shifting will also alter the step response tracking accuracy. These effects are quantified in Table 3-1. Note that, for each actuator configuration, as $\lambda_2$ increases the controlled system exhibits instability as one or more of the closed-loop poles moves outside the unit circle; but the range of $\lambda_2$ retaining stability is increased by each actuator configuration relative to (3-61). (The factorization scheme used is the same as that in Appendix C. See Appendix E for details.)

Table 3-2 summarizes the six examples that will be used to meet the objectives of the preceding section. The next section presents the adaptive control algorithms to be applied.

| actuator config. | $\lambda_2$ | poles #1 | #2,3 | #4,5 | #6,7 | zeros: #1 | #2,3 | #4,5 | DC gain |
|---|---|---|---|---|---|---|---|---|---|
| Fig. 3-12a | 1 | $0.850\angle 0^\circ$ | $0.890\angle \pm 18.4^\circ$ | $0.344\angle \pm 109.8^\circ$ | $0.950\angle \pm 143.9^\circ$ | $0.960\angle 180^\circ$ | $0.953\angle \pm 98.7^\circ$ | $0\angle \pm 0^\circ$ | 2.2 |
| | 10 | $0.865\angle 0^\circ$ | $0.896\angle \pm 18.2^\circ$ | $0.544\angle \pm 97.3^\circ$ | $0.964\angle \pm 149.3^\circ$ | $0.959\angle 180^\circ$ | $0.952\angle \pm 44.8^\circ$ | $0\angle \pm 0^\circ$ | 2.5 |
| | 20 | $0.877\angle 0^\circ$ | $0.902\angle \pm 17.9^\circ$ | $0.675\angle \pm 91.9^\circ$ | $0.997\angle \pm 153.2^\circ$ | $0.959\angle 180^\circ$ | $0.952\angle \pm 34.0^\circ$ | $0\angle \pm 0^\circ$ | 2.8 |
| | 50 | $0.906\angle 0^\circ$ | $0.915\angle \pm 17.4^\circ$ | $0.904\angle \pm 85.4^\circ$ | $1.080\angle \pm 159.7^\circ$ | $0.959\angle 180^\circ$ | $0.951\angle \pm 24.3^\circ$ | $0\angle \pm 0^\circ$ | $\infty$ |
| Fig. 3-12b | 1 | $0.820\angle 0^\circ$ | $0.868\angle \pm 19.3^\circ$ | $0.154\angle \pm 89.3^\circ$ | $0.950\angle \pm 143.8^\circ$ | $0.960\angle 180^\circ$ | $0.953\angle \pm 98.7^\circ$ | $0\angle \pm 0^\circ$ | 2.2 |
| | 10 | $0.841\angle 0^\circ$ | $0.881\angle \pm 18.8^\circ$ | $0.469\angle \pm 86.1^\circ$ | $0.963\angle \pm 148.6^\circ$ | $0.959\angle 180^\circ$ | $0.952\angle \pm 44.8^\circ$ | $0\angle \pm 0^\circ$ | 2.5 |
| | 20 | $0.862\angle 0^\circ$ | $0.889\angle \pm 18.4^\circ$ | $0.629\angle \pm 83.8^\circ$ | $0.992\angle \pm 152.3^\circ$ | $0.959\angle 180^\circ$ | $0.952\angle \pm 34.0^\circ$ | $0\angle \pm 0^\circ$ | 2.8 |
| | 50 | $0.897\angle 0^\circ$ | $0.907\angle \pm 17.7^\circ$ | $0.886\angle \pm 80.3^\circ$ | $1.070\angle \pm 158.7^\circ$ | $0.959\angle 180^\circ$ | $0.951\angle \pm 24.3^\circ$ | $0\angle \pm 0^\circ$ | $\infty$ |

TABLE 3-1: Pole-Zero Locations for Reduced-Order Control of Example 6 with Different Actuator Connections.

| Ex# | plant poles* | plant zero(s)* | DC gain | actuator pole* | actuator DC gain | desired pole(s)* | desired DC gain |
|---|---|---|---|---|---|---|---|
| 1 | 0.95,0.2 | 0.3 | 1.31 | - | - | 0.8 | 2 |
| 2 | 0.9,0.1 | 0.011 | 0.99 | - | - | 0.65 | 2 |
| 3 | 0.9,0.8 | 0.82 | 0.9 | - | - | 0.5 | 2 |
| 4 | 0.9,0.7 | 0.4 | 4.0 | - | - | 0.6 | 2 |
| 5 | $0.923 \pm j0.231$<br>$-0.762 \pm j0.570$ | $-0.960$<br>$-0.144 \pm j0.942$ | 4.04 | - | - | $0.844 \pm j0.171$ | 2.21 |
| 6 | $0.923 \pm j0.231$<br>$-0.762 \pm j0.570$ | $-0.960$<br>$-0.144 \pm j0.942$ | 4.04 | 0.607 | 1 | $0.844 \pm j0.171$ | 2.21 |

* All singularities are z-plane equivalents

TABLE 3-2: Examples Summary.

## IV. ADAPTIVE CONTROL ALGORITHMS

Presently, adaptive control schemes can be categorized by the following characteristics: (i) basis for parameter update scheme via gradient (G) search procedures or stability (S) theory analysis, (ii) formulation of parameter estimate error as equation (E) or output (O) error, and (iii) designation of parameters to be identified as plant parameters for indirect (I) adaptive control or as controller parameters for direct (D) adaptive control. Of the eight combinations only seven are presently available with GOD not yet developed. Of the remaining seven, two pairs are pairwise identical, GEI/SEI and GED/SED, due to the resulting equivalence of gradient and stability based equation error parameter estimation. The remaining five distinct classes providing adaptive solution to the model following problems of the pre ·ding section are detailed below with appropriate source reference. The designer selected variables of each are listed following each algorithm statement.

A reduced-order adaptive model-following problem requires specification of a time-varying feedback control law causing the output of the plant

$$y(k) = \sum_{j=1}^{\bar{m}} \bar{b}_j \, u(k-j) + \sum_{i=1}^{\bar{n}} \bar{a}_i \, y(k-i) \tag{4-1}$$

to asymptotically "follow" that of the stable reference model

$$s(k) = c \, r(k-1) + \sum_{i=1}^{n} a_i \, s(k-i) \tag{4-2}$$

without a priori knowledge of the plant parameters $\bar{a}_i$ and $\bar{b}_j$ or plant

dimensions $\bar{m}$ and $\bar{n}$ under the assumption that the plant is described by

$$y(k) = \sum_{j=1}^{m} b_j \, u(k-j) + \sum_{i=1}^{n} a_i \, y(k-i), \tag{4-3}$$

where $\bar{m} > m$ and $\bar{n} > n$. Only if $\bar{m} = m$, $\bar{n} = n$, $\bar{b}_j = b_j$, and $\bar{a}_i = a_i$ are the following solutions expected to cause $y \to s$. If, as in examples 5 and 6 of the preceding section, the model of (4-2) includes zeros, which are restricted to match those of (4-3), this converts the problem to one of adaptive pole-shifting. Since these zeros are identified in indirect adaptive control the pole-shifting objective presents no apparent difficulty, though no globally convergent, even nonreduced-order, indirect adaptive pole-shifting scheme has been proven due to sufficient excitation requirements [30] or matrix polynomial solution singularity problems [31]. Direct adaptive controllers must assume the availability of the plant numerator coefficients, i.e. the plant zeros, [32] or the stability of the plant [33] to achieve globally stable pole-shifting (or replacement). Therefore such direct adaptive control schemes (or those requiring numerator cancellation [17]) are not presented for examples 5 and 6. These restrictions on direct adaptive control support the statement that presently indirect adaptive control appears more suited to the ROAC problem requiring pole-shifting.

The algorithms, specialized for the examples of the preceding section and written in implementation sequence, are:

For Examples 1-4:

GEI/SEI [17], [21], [22], [34] - [36]:

$$e(k-1) = y(k-1) - \hat{a}(k-1)y(k-2) - \hat{b}(k-1)u(k-2) \tag{4-4}$$

$$\hat{a}(k) = \hat{a}(k-1) + \left[\frac{\mu}{h+\mu y^2(k-2)+\rho u^2(k-2)}\right] y(k-2)e(k-1) \tag{4-5}$$

$$\hat{b}(k) = \hat{b}(k-1) + \left[\frac{\rho}{h+\mu y^2(k-2)+\rho u^2(k-2)}\right] u(k-2)e(k-1) \tag{4-6}$$

$$\hat{g}(k-1) = \frac{c}{\hat{b}(k)} \tag{4-7}$$

$$\hat{f}(k-1) = \frac{d-\hat{a}(k)}{\hat{b}(k)} \tag{4-8}$$

Designer-selected variable requirements:

$$\mu > 0 \tag{4-9}$$

$$\rho > 0 \tag{4-10}$$

$$0 < h < 2 \tag{4-11}$$

$h$ is nominally chosen as one unless such a value would cause $\hat{b}(k) \approx 0$. Then $h$ is chosen within the range in (4-11) such that $|\hat{b}(k)| > \bar{\varepsilon} \approx 0^+$.

GOI [23], [25], [35] :

$$\hat{y}(k-1) = \hat{a}(k-1)\hat{y}(k-2) + \hat{b}(k-1)u(k-2) \tag{4-12}$$

$$\lambda(k-1) = \hat{y}(k-2) + \hat{a}(k-1)\lambda(k-2) \tag{4-13}$$

$$\gamma(k-1) = u(k-2) + \hat{a}(k-1)\gamma(k-2) \tag{4-14}$$

$$e(k-1) = y(k-1) - \hat{y}(k-1) \tag{4-15}$$

$$\hat{a}(k) = \hat{a}(k-1) + \left[\frac{\mu}{h+\mu\lambda^2(k-1)+\rho\gamma^2(k-1)}\right] \lambda(k-1)e(k-1) \tag{4-16}$$

$$\hat{b}(k) = \hat{b}(k-1) + \left[\frac{\rho}{h+\mu\lambda^2(k-1)+\rho\gamma^2(k-1)}\right] \gamma(k-1)e(k-1) \tag{4-17}$$

$$\hat{g}(k-1) = \frac{c}{\hat{b}(k)} \tag{4-18}$$

$$\hat{f}(k-1) = \frac{d-\hat{a}(k)}{\hat{b}(k)} \tag{4-19}$$

Designer-selected variable restrictions:

$$\mu > 0 \text{ and small} \tag{4-20}$$

$$\rho > 0 \text{ and small} \tag{4-21}$$

$$0 < h < 2 \tag{4-22}$$

h is nominally chosen as one unless such a value would cause $\hat{b}(k) \approx 0$. Then h is chosen within the range in (4-22) such that $|\hat{b}(k)| > \bar{\varepsilon} \approx 0^+$.

SOI: [8], [24], [35]:

$$\hat{y}(k-1) = \hat{a}(k-1)z(k-2) + \hat{b}(k-1)u(k-2) \tag{4-23}$$

$$v(k-1) = \frac{y(k-1) - \hat{y}(k-1) + q[y(k-2) - z(k-2)]}{h + \mu z^2(k-2) + \rho u^2(k-2)} \tag{4-24}$$

$$\hat{a}(k) = \hat{a}(k-1) + \mu z(k-2)v(k-1) \tag{4-25}$$

$$\hat{b}(k) = \hat{b}(k-1) + \rho u(k-2)v(k-1) \tag{4-26}$$

$$z(k-1) = \hat{a}(k)z(k-2) + \hat{b}(k)u(k-2) \tag{4-27}$$

$$\hat{g}(k-1) = \frac{c}{\hat{b}(k)} \tag{4-28}$$

$$\hat{f}(k-1) = \frac{d-\hat{a}(k)}{\hat{b}(k)} \tag{4-29}$$

Designer-selected variable restrictions:

$$\mu > 0 \tag{4-30}$$

$$\rho > 0 \tag{4-31}$$

$$\mathrm{Re}\left\{\frac{1+qz^{-1}}{1-a_1z^{-1}}\right\} > 0 \qquad \forall |z| = 1 \tag{4-32}$$

q can be chosen as zero for any stable $a_1$ and (4-32) will be satisfied.

$$1 \leq h < 2 \tag{4-33}$$

h is nominally chosen as one unless such a value would cause $\hat{b}(k) \approx 0$. Then h is chosen within the range in (4-33) such that $|\hat{b}(k)| > \bar{\varepsilon} \approx 0^+$.

GED/SED [17][37]:

$$v(k-1) = cr(k-2) + d\,y(k-2) - y(k-1) \tag{4-34}$$

$$\hat{g}(k-1) = \hat{g}(k-2) + \frac{\rho r(k-2)v(k-1)}{h[1+\mu y^2(k-2)+\rho u^2(k-2)]} \tag{4-35}$$

$$\hat{f}(k-1) = \hat{f}(k-2) + \frac{\mu y(k-2)v(k-1)}{h[1+\mu y^2(k-2)+\rho u^2(k-2)]} \tag{4-36}$$

Designer-selected variable restrictions

$$\mu > 0 \tag{4-37}$$

$$\rho > 0 \tag{4-38}$$

$$|h| > \frac{|b|}{2} \text{ and } \mathrm{sgn}(h \quad = \mathrm{sgn}(b) \tag{4-39}$$

SOD [17]-[20]:

$$\beta(k-1) = \rho r^2(k-2) + \mu y^2(k-2) \tag{4-40}$$

$$\gamma(k-1) = (d+q)h\beta(k-2)v(k-2) + d\gamma(k-2) \tag{4-41}$$

$$v(k-1) = (1+h\beta(k-1))^{-1}\{s(k-1)-y(k-1)+q[s(k-2)-y(k-2)]-\gamma(k-1)\} \tag{4-42}$$

$$\hat{g}(k-1) = \hat{g}(k-2) + \rho r(k-2)v(k-1) \tag{4-43}$$

$$\hat{f}(k-1) = \hat{f}(k-2) + \mu y(k-2)v(k-1) \tag{4-44}$$

**Designer-selected variable restrictions:**

$$\mu > 0 \tag{4-45}$$

$$\rho > 0 \tag{4-46}$$

$$|h| > \frac{|b|}{2} \text{ and } \mathrm{sgn}(h) = \mathrm{sgn}(b) \tag{4-47}$$

$$\mathrm{Re}\left\{\frac{1+qz^{-1}}{1-fz^{-1}}\right\} > 0 \qquad \forall |z| = 1 \tag{4-48}$$

This last condition is easily satisfied by $q = -d$, which should equate SOD to GED/SED [18].

Each of the preceding schemes (4-4) - (4-48) provides $\hat{g}(k-1)$ and $\hat{f}(k-1)$ for

$$u(k-1) = \hat{g}(k-1)r(k-1) + \hat{f}(k-1)y(k-1) \tag{4-49}$$

for application to (3-2) and generation of $y(k)$. Then these recursions are repeated.

For Examples 5 and 6:

GEI/SEI:

$$e(k-1) = y(k-1) - \hat{\alpha}_1(k-1)y(k-2) - \hat{\alpha}_2(k-2)y(k-3) - \hat{\beta}_1(k-1)u(k-2) - \hat{\beta}_2(k-2)u(k-3) \tag{4-50}$$

$$v(k-1) = e(k-1)/[h+\mu_1 y^2(k-2) + \mu_2 y^2(k-3) + \rho_1 u^2(k-2) + \rho_2 u^2(k-3)] \tag{4-51}$$

$$\hat{\alpha}_1(k) = \hat{\alpha}_1(k-1) + \mu_1 v(k-1)y(k-2) \tag{4-52}$$

$$\hat{\alpha}_2(k) = \hat{\alpha}_2(k-1) + \mu_2 v(k-1)y(k-3) \tag{4-53}$$

$$\hat{\beta}_1(k) = \hat{\beta}_1(k-1) + \rho_1 v(k-1)u(k-2) \tag{4-54}$$

$$\hat{\beta}_2(k) = \hat{\beta}_2(k-1) + \rho_2 v(k-1)u(k-3) \tag{4-55}$$

Designer-selected variable restrictions:

$$\mu_1 > 0 \tag{4-56}$$

$$\mu_2 > 0 \tag{4-57}$$

$$\rho_1 > 0 \tag{4-58}$$

$$\rho_2 > 0 \tag{4-59}$$

$$0 < h < 2 \tag{4-60}$$

h is nominally chosen as one unless such a value would cause $\hat{\beta}_2(k) \approx 0$ or $\hat{\beta}_1(k)\hat{\alpha}_1(k) + \hat{\beta}_2(k) - \hat{\alpha}_2(k)\hat{\beta}_1^2(k)/\hat{\beta}_2(k) \approx 0$. Then h is chosen within the range in (4-60) such that the absolute value of both terms $> \bar{\varepsilon} \approx 0^+$.

GOI:

$$\hat{y}(k-1) = \hat{\alpha}_1(k-1)\hat{y}(k-1) + \hat{\alpha}_2(k-1)\hat{y}(k-2) + \hat{\beta}_1(k-1)u(k-2) + \hat{\beta}_2(k-1)u(k-3) \tag{4-61}$$

$$\lambda_1(k-1) = \hat{y}(k-2) + \hat{\alpha}_1(k-1)\lambda_1(k-2) + \hat{\alpha}_2(k-1)\lambda_1(k-3) \tag{4-62}$$

$$\lambda_2(k-1) = \hat{y}(k-3) + \hat{\alpha}_1(k-1)\lambda_2(k-2) + \hat{\alpha}_2(k-1)\lambda_2(k-3) \tag{4-63}$$

$$\gamma_1(k-1) = u(k-2) + \hat{\alpha}_1(k-1)\gamma_1(k-2) + \hat{\alpha}_2(k-1)\gamma_1(k-3) \tag{4-64}$$

$$\gamma_2(k-1) = u(k-3) + \hat{\alpha}_1(k-1)\gamma_2(k-2) + \hat{\alpha}_2(k-1)\gamma_2(k-3) \tag{4-65}$$

$$v(k-1) = [y(k-1) - \hat{y}(k-1)]/[h + \mu_1\lambda_1^2(k-1) + \mu_2\lambda_2^2(k-1) + \rho_1\gamma_1^2(k-1) + \rho_2\gamma_2^2(k-1)] \tag{4-66}$$

$$\hat{\alpha}_1(k) = \hat{\alpha}_1(k-1) + \mu_1 v(k-1)\lambda_1(k-1) \tag{4-67}$$

$$\hat{\alpha}_2(k) = \hat{\alpha}_2(k-1) + \mu_2 v(k-1)\lambda_2(k-1) \tag{4-68}$$

$$\hat{\beta}_1(k) = \hat{\beta}_1(k-1) + \rho_1 v(k-1)\gamma_1(k-1) \tag{4-69}$$

$$\hat{\beta}_2(k) = \hat{\beta}_2(k-1) + \rho_2 v(k-1)\gamma_2(k-1) \tag{4-70}$$

**Designer-selected variable restrictions:**

$\mu_1 > 0$ and small (4-71)

$\mu_2 > 0$ and small (4-72)

$\rho_1 > 0$ and small (4-73)

$\rho_2 > 0$ and small (4-74)

$0 < h < 2$ (4-75)

h is nominally chosen as one unless such a value would cause $\hat{\beta}_2(k) \approx 0$ or $\hat{\beta}_1(k)\hat{\alpha}_1(k) + \hat{\beta}_2(k) - \hat{\alpha}_2(k)\hat{\beta}_1^2(k)/\hat{\beta}_2(k) \approx 0$. Then h is chosen within the range of (4-75) such that the absolute value of both terms $> \bar{\varepsilon} \approx 0^+$.

<u>SOI</u>:

$$\hat{y}(k-1) = \hat{\alpha}_1(k-1)z(k-2) + \hat{\alpha}_2(k-1)z(k-3) + \hat{\beta}_1(k-1)u(k-2) + \hat{\beta}_2(k-1)u(k-3) \tag{4-76}$$

$$v(k-1) = \frac{\{y(k-1)-\hat{y}(k-1)+q_1[y(k-2)-z(k-2)]+q_2[y(k-3)-z(k-3)]\}}{\{h+\mu_1 z^2(k-2)+\mu_2 z^2(k-3)+\rho_1 u^2(k-2)+\rho_2 u^2(k-3)\}} \tag{4-77}$$

$$\hat{\alpha}_1(k) = \hat{\alpha}_1(k-1) + \mu_1 v(k-1)z(k-2) \tag{4-78}$$

$$\hat{\alpha}_2(k) = \hat{\alpha}_2(k-1) + \mu_2 v(k-1)z(k-3) \tag{4-79}$$

$$\hat{\beta}_1(k) = \hat{\beta}_2(k-1) + \rho_1 v(k-1)u(k-2) \tag{4-80}$$

$$\hat{\beta}_2(k) = \hat{\beta}_2(k-1) + \rho_2 v(k-1)u(k-3) \tag{4-81}$$

$$z(k-1) = \hat{\alpha}_1(k)z(k-2) + \hat{\alpha}_2(k)z(k-3) + \hat{\beta}_1(k)u(k-2) + \hat{\beta}_2(k)u(k-3) \tag{4-82}$$

Designer-selected variable restrictions:

$\mu_1 > 0$ (4-83)

$\mu_2 > 0$ (4-84)

$\rho_1 > 0$ (4-85)

$\rho_2 > 0$ (4-86)

$$\operatorname{Re}\left\{ \frac{1 + q_1 z^{-1} + q_2 z^{-2}}{1 - \alpha_{11} z^{-1} - \alpha_{12} z^{-2}} \right\} > 0 \qquad \forall |z| = 1 \tag{4-87}$$

$q_1$ could be chosen as -0.98 and $q_2$ as zero for the rapidly sampled examples 5 and 6 and (4-87) would be satisfied.

$$1 \le h < 2 \tag{4-88}$$

h is nominally chosen as one unless such a value would cause $\hat{\beta}_2(k) \approx 0$ or $\hat{\beta}_1(k)\hat{\alpha}_1(k) + \hat{\beta}_2(k) - \hat{\alpha}_2(k)\hat{\beta}_1^2(k)/\hat{\beta}_2(k) \approx 0$. Then h is chosen within the range of (4-88) such that the absolute value of both terms $> \bar{\epsilon} > 0^+$.

Each of the last three algorithms in (4-50)-(4-88) provides $\hat{\alpha}_1(k)$, $\hat{\alpha}_2(k)$, $\hat{\beta}_1(k)$, $\hat{\beta}_2(k)$ for use in (refer to (3-27)-(3-30)).

$$\hat{\eta}_1(k-1) = \gamma_1 - \hat{\alpha}_1(k) \tag{4-89}$$

$$\hat{\nu}_1(k-1) = \frac{[\hat{\eta}_1(k-1)\hat{\alpha}_2(k)+(\hat{\eta}_1(k-1)\hat{\alpha}_1(k)-\hat{\alpha}_2(k)+\gamma_2)(\hat{\alpha}_1(k)-\hat{\alpha}_2(k)\hat{\beta}_1(k)/\hat{\beta}_2(k))]}{[\hat{\beta}_1(k)\hat{\alpha}_1(k)+\hat{\beta}_2(k)-\hat{\alpha}_2(k)\hat{\beta}_1^2(k)/\hat{\beta}_2(k)]} \tag{4-90}$$

$$\hat{\eta}_2(k-1) = \hat{\alpha}_1(k)\hat{\eta}_1(k-1) + \gamma_2 - \hat{\beta}_1(k)\hat{\nu}_1(k-1) - \hat{\alpha}_2(k) \tag{4-91}$$

$$\hat{\nu}_2(k-1) = \hat{\eta}_2(k-1)\hat{\alpha}_2(k)/\hat{\beta}_2(k) \tag{4-92}$$

for parameterization of

$$\begin{aligned} u(k-1) = {} & \delta r(k-1) + \hat{\eta}_1(k-1)u(k-2) + \hat{\eta}_2(k-1)u(k-3) \\ & + \hat{\nu}_1(k-1)y(k-2) + \hat{\nu}_2(k-1)y(k-3) \end{aligned} \tag{4-93}$$

for application to the difference equation representation of (3-17)

$$\begin{aligned} y(k) = {} & (\alpha_{21} + \alpha_{11})y(k-1) + (\alpha_{22} - \alpha_{11}\alpha_{21} + \alpha_{12})y(k-2) \\ & - (\alpha_{11}\alpha_{22} + \alpha_{12}\alpha_{21})y(k-3) - \alpha_{12}\alpha_{22}y(k-4) \\ & + (\beta_{11} + \beta_{21})u(k-1) - (\beta_{11}\alpha_{21} - \beta_{12} + \beta_{21}\alpha_{11} - \beta_{22})u(k-2) \\ & - (\beta_{11}\alpha_{22} + \beta_{12}\alpha_{21} + \beta_{21}\alpha_{12} + \beta_{22}\alpha_{11})u(k-3) \\ & - (\beta_{12}\alpha_{22} + \beta_{22}\alpha_{12})u(k-4) \end{aligned} \tag{4-94}$$

generating $y(k)$. Note that all of the preceding schemes in (4-4)-(4-93) use the current information as efficiently as possible and therefore result in causal rather than strictly causal adaptive controllers [17]. (See Appendix F for programs of each of these controllers.)

## V. TEST FORMATS

Having specified the non-adaptive and adaptive control strategies and objectives the remaining items to be selected for the testing of the guidelines in the second section are the reference inputs, adaption step-size constants, parameter estimate initializations, and performance measures.

Selection of the reference signal is important for two reasons: richness and magnitude. The magnitude of the reference signal is important due to the nonlinearity of the adaptive control system and therefore the anticipated transient (at least) differences due to different signal levels. The richness is important because it is well-recognized [38] that a sufficient "input" richness is necessary to perturb all of the modes of the system for their "identification." This "input" differs for the indirect and direct approaches and the equation and output error formulations. For the equation error indirect approach the plant input and output must be sufficiently rich. Since the plant is linear this translates to a sufficient frequency content of the plant input (or control) signal, of which the reference is only one component. The difficulty of assuming adaptive control input richness based on reference signal richness is addressed in [30]. For the output error indirect approach the plant input and identifier output must be sufficiently rich. In an open-loop output error identification task the identifier output richness can be translated [39] solely to input richness necessity under the assumption that the identifier output asymptotically converges to the plant output, which requires sufficient identifier order absent from ROAC. For direct adaptive control the control parameter identification problem must be

recast as an open-loop identification task as in [19] and [37]. In the equation error case the richness requirements are converted principally to conditions on the model output to be tracked or, if generated by a linear, time-invariant model, to the forcing or reference signal [39]. Once again this assumes sufficient controller order. The requirements for output error based direct adaptive control, though as yet unspecified, are assumed to be similar.

All of the preceding richness requirements are for complete identification and assume non-reduced adaptive model order. As is well-recognized [22] [34] [37], for a control objective complete identification may not be necessary. Consider the control of any order system with a single constant output feedback gain and a reference signal gain. To achieve asymptotic convergence to a particular DC level set-point any feedback gain stabilizing the system in conjunction with the appropriate reference gain will be adequate. This applies readily to examples 1-4 in section three. As noted in [40] for open-loop output error (and by implication equation error) identification the richness of the input signal for stable reduced-order identification is dependent on the reduced-order model dimension, not that of the higher order plant. From a frequency response point of view, the viewpoint promoted in [41] for reduced-order modelling, the minimum number of sinusoids providing this richness matches the number of points at which the controller can specify the steady-state frequency response. Therefore, if this reduced-order controller stabilizes the system, asymptotically convergent steady-state control seems possible up to the richness level of the reduced-order model. When the signal to be tracked (or equivalently the reference signal) has a higher frequency content, reduced-order tracking will only be approximate at best. The reference

signal should be chosen to test this boundary condition, with the principal case of interest being reference signal over-richness. Therefore the following reference signals (all zero prior to $k = 0$) will be considered:

INPUT 1 (unit step input):

$$r_1(k) = 1 \tag{5-1}$$

INPUT 2 (single low-frequency sinusoid):

$$r_2(k) = 2 \sin(k\omega_2 T) \tag{5-2}$$

Note that the sample period T is assumed to be one second for examples 1-4. The frequency $\omega_2$ is chosen as 20% below the desired cutoff frequency for examples 1-4, i.e. 0.17851, 0.34463, 0.55452, and 0.40866 radians per second, respectively, and 20% below the dominant natural frequency for examples 5 and 6, i.e. 0.40 radians per second, with T = 0,5 seconds,

INPUT 3 (single high-frequency sinusoid):

$$r_3(k) = 2 \sin(k\omega_3 T) \tag{5-3}$$

Again the sample period is assumed to be one second for examples 1-4. The frequency $\omega_3$ is double the desired cutoff for examples 1-4, i.e. respectively 0.44629, 0.86157, 1.38629, and 1.02170, and double the dominant natural frequency for examples 5 and 6, i.e. 1.00, with T = 0,5 seconds,

INPUT 4 (non-zero mean, uniform, white noise):

$$r_4(k) \in [-0.5, 1.5] \tag{5-4}$$

This is clearly the over-rich input of most interest.

Since the rate of convergence and parameter estimate variance near convergence are commonly acknowledged as being affected by the adaptive step-size (SS) coefficients $\mu_i$ and $\rho_j$, two cases will be considered.

$SS_1 = 1$:

$$\mu_i = \rho_j = 1 \tag{5-5}$$

$SS_2 = 0.1$:

$$\mu_i = \rho_j = 0.1. \tag{5-6}$$

Given the approximate nature of the gradient formulas for GOI embodied in the required $\mu$ and $\rho$ smallness noted in (4-20)-(4-21) and (4-71)-(4-74) the $\mu$ and $\rho$ of (5-5) may lead to unstable behavior for GOI while (5-6) should prove more satisfactory. (Note that h in the normalizing term of all of the algorithms is selected as unity).

For output error type algorithms such as SOI and SOD the error smoothing coefficients q in (4-24), (4-47), and (4-77) are to be selected in satisfaction of (4-32), (4-48), and (4-87), respectively. As noted in [42] selection of these q to make (4-32), (4-48), and (4-87) equal unity does not offer the most rapid convergence. Furthermore, in a reduced-order application these q can influence the mean convergence point. Such factors may be used to advantage in (4-48) where f is known a priori, but in (4-32) and (4-87) the strictly positive real (SPR) condition is dependent on unknown plant parameters. Therefore several values will be tested. They are summarized in Table 5-1.

Since only local convergence is currently anticipated three settings of the initial plant (and therefore controller) parameter estimates (EST) will be considered corresponding to the nominal values from control design based on neglecting the second mode (i.e. $\varepsilon = 0$ for examples 1-4 and $\lambda_2 = 0$ for examples 5 and 6) and ± 20% of these values. These values are summarized in Table 5-2. The controller parameter initializations for the direct schemes are based on the corresponding plant parameter initializations and

| Example(s) | Algorithm | Error Smoothing Coefficient(s) | Label |
|---|---|---|---|
| 1 | SOI | -0.9 | SC- |
| | | -0.95 | SC0 |
| | | -0.97 | SC+ |
| | SOD | -0.7 | SC- |
| | | -0.8 | SC0 |
| | | -0.9 | SC+ |
| 2 | SOI | -0.9 | SC0 |
| | SOD | -0.65 | SC0 |
| 3 | SOI | -0.9 | SC0 |
| | SOD | -0.5 | SC0 |
| 4 | SOI | -0.8 | SC- |
| | | -0.9 | SC0 |
| | | -0.95 | SC+ |
| | SOD | -0.4 | SC- |
| | | -0.6 | SC0 |
| | | -0.8 | SC+ |
| 5&6 | SOI | $q_1=-0.9, q_2=0$ | SC- |
| | | $q_1=-1.846, q_2=0.9515$ | SC0 |
| | | $q_1=-0.96, q_2=0$ | SC+ |

TABLE 5-1: Error Smoothing Coefficients

| Example | Parameter | EST = +20% | EST = 0 | EST = -20% |
|---|---|---|---|---|
| 1 | $\hat{b}(0)$ | 0.078 | 0.065 | 0.052 |
| | $\hat{a}_1(0)$ | 1.14 | 0.95 | 0.76 |
| 2 | $\hat{b}(0)$ | 0.12 | 0.1 | 0.08 |
| | $\hat{a}_1(0)$ | 1.08 | 0.9 | 0.72 |
| 3 | $\hat{b}(0)$ | 0.096 | 0.08 | 0.064 |
| | $\hat{a}_1(0)$ | 1.08 | 0.9 | 0.72 |
| 4 | $\hat{b}(0)$ | 0.6 | 0.5 | 0.4 |
| | $\hat{a}_1(0)$ | 1.08 | 0.9 | 0.72 |
| 5&6 | $\hat{\beta}_{11}(0)$ | 0.144 | 0.120 | 0.096 |
| | $\hat{\beta}_{12}(0)$ | 0.139 | 0.116 | 0.093 |
| | $\hat{\alpha}_{11}(0)$ | 2.215 | 1.846 | 1.477 |
| | $\hat{\alpha}_{12}(0)$ | -1.086 | -0.905 | -0.724 |

TABLE 5-2: Parameter Estimate Initializations.

(3-8) and (3-9) for examples 1-4 and (3-27)-(3-30) for examples 5 and 6. Note the instability of the EST = + 20% initializations.

Since the control objective of all of the examples in section 3 is model-following, the tracking error is the principal performance measure. For the expectation of asymptotic tracking, the figures of merit should be divisible into short and long term quantities. The mean and variance of the tracking errors should be normalized and tabulated for comparison among the examples and between the fixed and adaptive control schemes. Similarly the input cost and controller (and plant, for indirect schemes) parameter estimates should be observed. Therefore the following quantities will be tabulated at $k_i$ = 1, 2, 5, 10, 20, 50, 100, 200, 500, and 1000 iterations.

Instantaneous desired output:

$$\mathrm{IDO}(k) = s(k) \tag{5-7}$$

Instantaneous controlled output:

$$\mathrm{IO}(k) = y(k) \tag{5-8}$$

Normalized instantaneous tracking error:

$$\mathrm{NITE}(k) = \frac{(s(k) - y(k))}{s(k)} \tag{5-9}$$

Normalized average segmented tracking error:

$$\mathrm{NASTE} = \left(\frac{1}{k_i - k_{i-1}}\right) \sum_{j=k_{i-1}+1}^{k_i} \mathrm{NITE}(j) \tag{5-10}$$

Normalized segmented tracking error variance

$$\mathrm{NSTEV} = \left(\frac{1}{n(n-1)}\right) \left\{ n \sum_{j=k_{i-1}+1}^{k_i} \mathrm{NITE}^2(j) - \left[ \sum_{j=k_{i-1}+1}^{k_i} \mathrm{NITE}(j) \right]^2 \right\} \tag{5-11}$$

(Note that n is $k_i - k_{i-1}$ minus the number of times $s(k) < 10^{-3}$, which are excluded due to (5-9). If NSTEV is negative due to round-off, it is printed as zero.)

Segmented average squared input

$$\text{SASI} = \left(\frac{1}{k_i - k_{i-1}}\right) \sum_{j=k_{i-1}+1}^{k_i} u^2(j) \tag{5-12}$$

Segmented average controller parameter estimate for f

$$\text{SACPE F} = \left(\frac{1}{k_i - k_{i-1}}\right) \sum_{j=k_{i-1}+1}^{k_i} \hat{f}(k) \tag{5-13}$$

(Similarly for g, a, b, α, β, ν, and η.)

Segmented controller parameter estimate variance for g

$$\text{SCPEV G} = \left\{ \frac{1}{(k_i - k_{i-1})(k_i - k_{i-1} - 1)} \right\} \left\{ (k_i - k_{i-1}) \sum_{j=k_{i-1}+1}^{k_i} \hat{f}^2(j) - \left[ \sum_{j=k_{i-1}+1}^{k_i} \hat{f}(j) \right]^2 \right\} \tag{5-14}$$

(Similarly for g, a, b, α, β, ν, and η.)

These quantities are tabulated for each of the combinations of example, adaptive control algorithm, reference input, step-size weights, error smoothing coefficients (if necessary), and parameter estimate initialization. (See Appendix F for listings of the programs compiling these figures of merit.) Table 5-3 summarizes all the various combinations that were simulated.

| Example | Control Alg. | Inputs | Par. Est. Init. | Step Size | Error Smooth Coeff. | Actuator Config-uration | Total |
|---|---|---|---|---|---|---|---|
| 1 | nonadaptive | all 4 | EST = 0 | zero | - | - | 4 |
| | GEI/SEI (1) | all 4 | all 3 | both | - | - | 24 |
| | GOI (2) | all 4 | all 3 | both | - | - | 24 |
| | SOI (3) | all 4 | all 3 | both | all 3 | - | 72 |
| | GED/SED (4) | all 4 | all 3 | both | - | - | 24 |
| | SOD (5) | all 4 | all 3 | both | all 3 | - | 72 |
| 2 | nonadaptive | all 4 | EST = 0 | zero | - | - | 4 |
| | GEI/SEI (1) | all 4 | EST = 0 | $SS_1$ | - | - | 4 |
| | GOI (2) | all 4 | EST = 0 | $SS_1$ | - | - | 4 |
| | SOI (3) | all 4 | EST = 0 | $SS_1$ | SCO | - | 4 |
| | GED/SED (4) | all 4 | EST = 0 | $SS_1$ | - | - | 4 |
| | SOD (5) | all 4 | EST = 0 | $SS_1$ | SCO | - | 4 |
| 3 | nonadaptive | all 4 | EST = 0 | zero | - | - | 4 |
| | GEI/SEI (1) | all 4 | EST = 0 | $SS_1$ | - | - | 4 |
| | GOI (2) | all 4 | EST = 0 | $SS_1$ | - | - | 4 |
| | SOI (3) | all 4 | EST = 0 | $SS_1$ | SCO | - | 4 |
| | GED/SED (4) | all 4 | EST = 0 | $SS_1$ | - | - | 4 |
| | SOD (5) | all 4 | EST = 0 | $SS_1$ | SCO | - | 4 |
| 4 | nonadaptive | all 4 | EST = 0 | zero | - | - | 4 |
| | GEI/SEI (1) | all 4 | all 3 | both | - | - | 24 |
| | GOI (2) | all 4 | all 3 | both | - | - | 24 |
| | SOI (3) | all 4 | all 3 | both | all 3 | - | 72 |
| | GED/SED (4) | all 4 | all 3 | both | - | - | 24 |
| | SOD (5) | all 4 | all 3 | both | all 3 | - | 72 |
| 5 | nonadaptive | all 4 | all 3 | zero | - | - | 12 |
| | GEI/SEI (1) | all 4 | all 3 | both | - | - | 24 |
| | GOI (2) | all 4 | all 3 | both | - | - | 24 |
| | SOI (3) | all 4 | all 3 | both | all 3 | - | 72 |
| 6 | nonadaptive | all 4 | all 3 | zero | - | Fig.12a&b | 24 |
| | GEI/SEI (1) | all 4 | all 3 | both | - | all 3 | 72 |
| | GOI (2) | all 4 | all 3 | both | - | all 3 | 72 |
| | SOI (3) | all 4 | all 3 | both | all 3 | all 3 | 216 |
| | | | | | | | 1008 |

TABLE 5-3: Tested Combinations.

# VI. RESULTS

Due to the number (1008) of printouts for the examples shown in Table 5-3 the tabulated results are under separate cover. (See Appendix F for pertinent program listings.) The next section provides an evaluation of these numerical results.

# VII. TEST RESULTS INTERPRETATIONS

Refer to section 2 for the statement of the objectives of this study. The four questions raised in that section will be addressed in this section with respect to the simulations of Section 6.

QUESTION (i):

Are heavily damped open-loop modes neglectable relative to more lightly damped modes? In particular:

a) Are the accepted criteria for neglecting modes, based on relative damping, realistic for the examples considered here in a fixed (non-adaptive) reduced-order controller?

b) For these examples, how do the various adaptive control schemes fare under the reduced order modeling "rules" concerning relative damping? More specific questions might be: Do the reduced-order adaptive control systems remain stable? Can the adaptive mechanism compensate for mis-modeling to yield improved performance over non-adaptive (modal) controllers of the same order?

In example 1, the open loop and closed loop modes are separable based on the accepted criteria for relative damping, so the good closed-loop (non-adaptive) reduced-order performance was to be expected (despite the fact that in this example and all the others the reduced-order modal model was based strictly on the partial fraction expansion term without DC gain correction). Example 2 represents a marginally separable system in closed loop, yet the (non-adaptive) reduced-order controller tracking errors are only slightly worse than in example 1. This indicates that, for these examples, this boundary between separable and inseparable

modes is reasonable but fuzzy (as expected).

Example 3 possesses neither closed loop nor open loop separability, but a pole-zero near-cancellation should leave one mode dominating the system bahavior. As shown in the simulation, the reduced-order modeling causes significant errors. This is presumably due to the relative closeness of the "cancelled" pole to the unit circle compared to the "dominant" pole. Small errors in cancellation in this case may have large effects on system behavior and the performance of reduced-order control.

The approximation of a pair of complex poles by a simple real axis pole is studied in example 4. Here, no first order model can be found to closely represent this second order system, and considerable degradation in performance by reduced order control should be expected. The simulations show this to be true; the steady state tracking errors are significant.

Of the five adaptive control algorithms, all but the second one, the gradient-output-error-indirect (GOI) scheme, were stable. In example one, which has the most accurate reduced order model, a larger number of test combinations of inputs, step sizes, and initial parameter estimates produce stable responses than the other examples. Even so, the only consistent results were for zero initial parameter error (with respect to the extracted modal model) and the step and random inputs. In these cases, the step response is improved over the non-adaptive case to essentially zero steady state error, while the random input causes a stable yet more widely ranging response. The other four adaptive algorithms produced essentially equivalent responses for most inputs, step sizes and smoothing coefficients. Parameter convergence occurred between 50 and 500 iterations for

most runs, the random input case being consistently faster. Also, the error between the desired and actual outputs always resulted in improvement over the non-adaptive output error. Some minor differences in responses were:

- Example 3 usually responded faster, due to the dominant pole being more highly damped.
- Algorithm 5 (stability-output-error-direct, SOD) produced consistently smaller steady state output tracking errors.
- Smaller step sizes improved the steady state tracking error variance, but lengthened convergence times.
- The smoothing coefficient SC- in algorithm 3 produced more accurate steady state tracking than the other smoothing coefficients.
- The adaptive parameter estimates have higher variances for example 2 than example 1 indicating more active searching due to the marginal dominance of the desired reduced-order objective in the problem setup.

Except for the GOI algorithm (#2), all adaptive schemes provided improved output tracking of desired models over non-adaptive control in examples 1-4. This is the result of the adaptive algorithms inclination to select some controller parameterization to "better fit" some control objective. In the indirect algorithms, the identified plant often has parameters that do not match either those of the actual plant or those of the assumed reduced order (modal) model. These effects are explored further in the discussion of the third question, on choosing the "best" reduced order model.

QUESTION (ii):

If actuator dynamics are included in the system such that the residual modes have frequencies within the stop band of the actuator frequency response, does improved reduced-order control result?

Examples 5 and 6 were used to test the effects of actuator inclusion. Example 5 included no actuator dynamics and was seen to be unstable for a large number of runs, particularly those with initial parameter estimate errors relative to the dominant mode. Example 6 included the three actuator configurations shown in Figure 3-12, where actuator configuration 2 in the simulations corresponded to actuator (a) in the figure, 3 to (b), and 4 to (c).

In the non-adaptive simulations, inclusion of an actuator provided improved tracking for those runs of example 5 that were stable based on the reduced-order modal model. For the fixed (initial) reduced order model parameter estimates that resulted in an unstable control system, actuator inclusion generally did not provide stability. Actuator configurations 2 and 4 provided up to an order of magnitude improvement in tracking compared to 3 and 1 (no actuator), which were similar in response.

Under adaption, the most improvement was obtained, conversely, by including an actuator as in configuration 3. This arrangement produced stable simulations for all but a few runs, where the output was apparently unstable but still bounded within the 1000 iterations tested. For the stable adaptive usage of actuator configuration 3, nearly all cases provided between 1 and 3 orders of magnitude improvement in tracking error over the other three actuator configurations.

Actuator arrangements 2 and 4 behaved almost identically during adaption under all conditions. They were seen to improve the response

over that without actuator dynamics for most runs, but occasionally their inclusion worsened the output tracking. The only consistent improvement over configuration 3 was for input 3, the high frequency sinusoid, where an order of magnitude difference was sometimes seen.

Among the three algorithms used on examples 5 and 6, the first (GED/SED) and third (GEI) behaved essentially the same, giving the responses discussed above. Algorithm 2 (GOI) caused unstable responses for all combinations of inputs, step sizes, initial parameter estimates, and actuator configurations.

For these examples, inclusion of a finite bandwidth actuator generally improved the performance of the reduced-order control. However, there were some combinations of inputs, initial parameter estimates, and adaptive step sizes that caused poorer behavior with an actuator than without. The actuator arrangement that allowed both the plant identifier and controller to sample the actuator output was clearly the best choice for the majority of cases here. This structure effectively includes the actuator as part of the controller. As noted in section 3, this is not the most physically feasible configuration since the actuator outputs can be inaccessible.

QUESTION (iii):

Does an "optimal" reduced order controller result from retaining the dominant modes of the plant for the reduced-order plant model, or by choosing a model (or controller) parameterization that does not correspond to plant modes but is based on matching (in some sense) a reduced order control objective?

As phrased, the latter alternative intuitively seems "better". This question can be answered for these examples by comparing the "zero initial parameter estimate error" parameters with the steady state parameters

in the adaptive runs. If the difference in parameters is slight, then the selection of the dominant plant mode for the control model would seem to be a reasonable choice with respect to the adaptive parameter estimator update criterion. If the parameter difference is considerable and the adaptive responses are deemed better than the non-adaptive ones , then a reduced order model based on those parameters selected by the "behavior approximating" adaptive algorithms would seem the better choice.

As would be expected, the adaptive runs showing the greatest improvement over the fixed controller runs were those with the largest differences in initial and steady state parameters. The implication is that choosing dominant modes out of the set of plant modes for the reduced order model is often not as "optimal" as choosing model "modes" based on some performance criterion.

QUESTION (iv):

Are indirect and direct adaptive controllers equivalent and interchangeable in reduced-order application?

The absurdity of such an expectation is underscored by a comparison of the algorithms in section IV. Note that indirect schemes involve division by estimated quantities as in, e.g., (4-7) and (4-8), but that direct schemes do not. Clearly, if the estimated quantities ever approach zero the resulting quotient is quite large. A similar mechanism does not seem to appear in the direct algorithms. The claims of equivalence seem to rest on the (linear?) transformability between the indirect and direct approaches in simple cases, such as inverse or model-following control. If the algorithms are not identical then, as clearly demonstrated by this study, the transient differences in the full order case translate into steady-state differences in the reduced-order case. Unfortunately neither indirect or direct adaptive control seems consistently superior to the other in all situations.

# VIII. CONCLUSION

The bulk of the data generated from the 1008 simulations attests to the difficulty of drawing definitive conclusions from such simulation studies. This is further complicated for studies of adaptive controllers by the number of variables, e.g. plant parameters, control objective, initial parameter estimates, step-size weights, error smoothing coefficients and input richness, effecting overall performance. Conclusively examining all possibilities is clearly unfeasible. Even a focused study, such as this one with respect to the four objectives commented on in the preceding section, can only be expected to provide heavily qualified remarks. At best, such a study can only be used to test and augment the "folklore". As particular effects are isolated without contradiction further simulation studies are in order to test seeming generalizations. The ones that survive closer scrutiny provide guideposts for restricted analytical studies, that could prove equally fruitless if not initially well-focused.

The data generated by this study may provide a database for judging eventual conclusions of future studies. A different format, e.g. graphical, for even the tests run here might facilitate evaluation of different conclusions. The one general (tautological) conclusion that seems well documented by this report is that all present adaptive control schemes will fail in <u>some</u> reduced-order application. Though most of the (stable) adaptive controllers tested seem to better the performance of fixed modal controllers this does <u>not</u> present a conclusive argument for unsupervised adaptive control. It is trite but true that further study is necessary.

Acknowledgement: Several helpful discussions with M. J. Balas of Rensselaer Polytechnic Institute and R. E. Skelton of Purdue University proved indispensable in the development of Sections I and II of this report.

# IX. REFERENCES


[1] M. J. Balas and C. R. Johnson, Jr., "Adaptive control of distributed parameter systems: The ultimate reduced order problem," Proc. 18th IEEE Conf. on Dec. and Control, Ft. Lauderdale, FL, pp. 1013-1017, December 1979.

[2] G. C. Goodwin and K. S. Sin, "Effect of model, system, and controller order on adaptive control," Proc. 18th IEEE Conf. on Dec. and Control, Ft. Lauderdale, FL, pp. 202-206, December 1979.

[3] C. R. Johnson, Jr., "Adaptive modal control of large flexible spacecraft," Journal of Guidance and Control, vol. 3, no. 4, July-August 1980.

[4] G. C. Goodwin and P. J. Ramadge, "Design of restricted complexity adaptive regulators," IEEE Trans. on Auto. Control, vol. AC-24, no.4, pp. 584-588, August 1979.

[5] R. Genesio and M. Milanese, "A note on the derivation and use of reduced-order models," IEEE Trans. on Auto. Control, vol. AC-21, no.1, pp. 118-122, February 1976.

[6] R. E. Skelton and C. Z. Gregory, "Measurement feedback and model reduction by modal cost analysis," Proc. 1979 Jt. Aut. Control Conf., Denver, CO, pp. 211-219, 1979.

[7] R. E. Skelton, "Cost decomposition of linear systems with applications to model reduction," Int. J. Control, to appear.

[8] Y. D. Landau, Adaptive Control: The Model Reference Approach, New York: Marcel-Dekker, 1979.

[9] C. R. Johnson, Jr. and R. C. Montgomery, "A distributed system adaptive control strategy," IEEE Trans. on Aerospace and Electr. Sys., vol. AES-15, no. 5, pp. 601-612, September 1979.

[10] M. J. Balas and C. R. Johnson, Jr., "Toward adaptive control of large space structures," in Applications of Adaptive Control, eds. K. S. Narendra and R. V. Monopoli. New York: Academic Press, pp.313-344, 1980.

[11] C. R. Johnson, Jr. and M. J. Balas, "Distributed parameter system coupled ARMA expansion identification and adaptive parallel IIR filtering: A unified problem statement," Proc. 13th Asilomar Conf. on Circuits, Sys., and Computers, Pacific Grove, CA, pp. 219-225, November 1979.

[12] R. C. Dorf, Modern Control Systems. Reading, MA: Addison-Wesley, p. 184, 1980.

[13] K. Ogata, Modern Control Engineering. Englewood Cliffs, NJ: Prentice-Hall, 1970.

[14] B. F. Gardner, Jr. and J. B. Cruz, Jr., "Lower order control for systems with fast and slow modes," Automatica, vol. 16, no.2, pp. 211-213, March 1980.

[15] P. C. Hughes and T. M. Abdel-Rahman, "Stability of proportional-plus-derivative-plus-integral control of flexible spacecraft," Journal of Guidance and Control, vol. 2, [illegible] pp. 499-503, November-December 1979.

[16] K. S. Narendra and L. S. Valavani, "Direct and indirect model reference adaptive control," Automatica, vol. 15, no.6, pp. 653-664, November 1979.

[17] C. R. Johnson, Jr., "Input matching, error augmentation, self-tuning, and output error identification: Algorithmic similarities in discrete adaptive model following," IEEE Trans. on Auto. Control, vol. AC-25, no.4, pp.697-703, August 1980.

[18] C. R. Johnson, Jr., "Explication of a simple output error based model reference adaptive controller," VPI&SU Dept. of Elec. Eng. Tech. Report No. EE8013, June 1980.

[19] C. R. Johnson, Jr., "An output error identification interpretation of model reference adaptive control," Automatica, vol. 16, no.4, pp. 419-421, July 1980.

[20] K. S. Narendra and Y-H. Lin, "Stable discrete adaptive control," IEEE Trans. on Auto. Control, vol. AC-25, no.3, pp. 456-461, June 1980.

[21] B. Widrow, J. M. McCool, M. G. Larimore, and C. R. Johnson, Jr., "Stationary and nonstationary learning characteristics of the LMS adaptive filter," Proc. IEEE, vol. 64, no.8, pp. 1151-1162, August 1976.

[22] C. R. Johnson, Jr., "Adaptive parameter matrix and output vector estimation via an equation error formulation," IEEE Trans. on Sys., Man, and Cybernetics, vol. SMC-9, no.7, pp. 392-397, July 1979.

[23] D. Parikh and N. Ahmed, "On an adaptive algorithm for IIR filters," Proc. IEEE, vol.66, no.5, pp. 585-587, May 1978.

[24] C. R. Johnson, Jr., "A convergence proof for a hyperstable adaptive recursive filter," IEEE Trans. on Info. Thy., vol. IT-25, no.6, pp. 745-749, November 1979.

[25] C. R. Johnson, Jr., "A stable family of adaptive IIR filters," Proc. 1980 IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, Denver, CO, pp. 997-1000, April 1980.

[26] C. R. Johnson, Jr. and M. J. Balas, "Reduced order adaptive controller studies," Proc. 1980 Jt. Auto. Control Conf., San Francisco, CA, Paper WP2-D, August 1980.

[27] C. P. Newman and C. S. Baradello, "Digital transfer functions for microcomputer control," IEEE Trans. on Sys., Man, and Cybernetics, vol. SMC-9, no. 12, pp. 856-860, December 1979.

[28] C. R. Johnson, Jr., "On direct adaptive control of nonminimum-phase plants," Information Sciences, vol. 20, pp. 165-179, April 1980.

[29] G. F. Franklin and J. D. Powell, Digital Control of Dynamic Systems. Reading, MA: Addison-Wesley, p. 64, 1980.

[30] G. Kreisselmeier, "Adaptive control via adaptive observation and asymptotic feedback matrix synthesis," IEEE Trans. on Auto. Control, vol. AC-25, no.4, pp. 717-722, August 1980.

[31] G. C. Goodwin and K. S. Sin, "Adaptive control of nonminimum phase systems," Univ. of Newcastle Dept. of Elec. Eng. Tech. Report No. EE 7918, August 1979.

[32] A. L. Hamm and C. R. Johnson, Jr., "Model reference adaptive control using an adjustable model," Proc. 12th Southeastern Symp. on Sys. Thy., Virginia Beach, VA, pp. 269-273, May 1980.

[33] C. R. Johnson, Jr., "An adjustable model reference adaptive control reconfiguration of self-tuning pole replacement," Proc. IEEE, vol. 68, no.2, pp. 295-296, February 1980.

[34] J. M. Mendel, Discrete Techniques of Parameter Estimation: The Equation Error Formulation. New York: Marcel Dekker, 1973.

[35] K. J. Åström, U. Borisson, L. Ljung, and B. Wittenmark, "Theory and application of self-tuning regulators," Automatica, vol. 13, no.5, pp. 507-517, September 1977.

[36] G. C. Goodwin, P. J. Ramadge, and P. E. Caines," Discrete-time multivariable adaptive control," IEEE Trans. on Auto. Control, vol. AC-25, no.3, pp. 449-456, June 1980.

[37] C. R. Johnson, Jr. and E. Tse, "Adaptive implementation of one-step-ahead optimal control via input matching," IEEE Trans. on Auto. Control, vol. AC-23, no.5, pp. 865-872, October 1978.

[38] J. S.-C. Yuan and W. M. Wonham, "Probing signals for model reference identification," IEEE Trans. on Auto. Control, vol. AC-22, no.4, pp. 530-538, August 1977.

[39] B. D. O. Anderson and C. R. Johnson, Jr., "Exponential convergence of adaptive identification and control algorithms," University of Newcastle Dept. of Elec. Eng. Tech. Report, April 1980.

[40] C. R. Johnson, Jr. and B. D. O. Anderson, "On reduced-order adaptive output error identification and adaptive IIR filtering," May 1980.

[41] S. A. Marshall, "The design of reduced-order systems," Int. J. Control, vol. 31, no.4, pp. 677-690, April 1980.

[42] C. R. Johnson, Jr. and T. Taylor, "CHARF convergence studies," Proc. 13th Asilomar Conf. on Circuits, Sys., and Computers, Pacific Grove, CA, pp. 403-407, November 1979.

## Appendix A: Root Locus Program Listings and Tabular Outputs

Contents:

```
$JOB          WATFIV,KP=29,PAGE=100,TIME=100
C       RTL1:
C       ROOTLOCUS INTERPRETATION OF THE UNADAPTED PLANT.
C       T.F.:Y(Z)/F(Z)=((B+E)*Z-(B*A2+E*A1))/(Z**2-(A1+A2+F*(B+E))*Z+A1*A
C       2+F*(B*A2+E*A1)
        DIMENSION E(4),XX1(500),XX2(500),YY1(500),YY2(500),AZ(500),F2(50)
        DIMENSION A1(4),A2(4),B(4),YM1(500),YM2(500)
        COMPLEX GC,FC,ZC(2)
        DATA E/0.01,-0.01,0.02,-0.3/
        DATA A1/0.95,0.9,0.9,0.9/
        DATA A2/0.2,0.1,0.8,0.7/
        DATA B/0.065,0.1,0.08,0.5/
        DO 300 J=1,4
        WRITE(6,200)A1(J),A2(J),B(J)
 200    FORMAT(1X,'A1=',F13.6,5X,'A2=',F13.6,5X,'B=',F13.6)
        WRITE(6,400)E(J)
 400    FORMAT(//,'                                              E=',F13.6)
        DO 100 K=1,50
        IF(J.EQ.4) GO TO 13
        F2(K)=-0.2*(K-1)
        GO TO 1
 13     F2(K)=0.2*(K-1)
 1      G=(A1(J)+A2(J)+F2(K)*(B(J)+E(J)))
        F=(A1(J)*A2(J)+F2(K)*(B(J)*A2(J)+E(J)*A1(J)))
        GC=CMPLX(G,0.0)
        FC=CMPLX(F,0.0)
        ZC(1)=(GC+CSQRT(GC**2-4.0*FC))/2.0
        ZC(2)=(GC-CSQRT(GC**2-4.0*FC))/2.0
        ZR1=REAL(ZC(1))
        ZR2=REAL(ZC(2))
        ZI1=AIMAG(ZC(1))
        ZI2=AIMAG(ZC(2))
        XX1(K)=ZR1
        XX2(K)=ZR2
        YY2(K)=ZI2
        YY1(K)=ZI1
        YMAG1=SQRT(ZR1**2+ZI1**2)
        YMAG2=SQRT(ZR2**2+ZI2**2)
        YM1(K)=YMAG1
        YM2(K)=YMAG2
        V=B(J)+E(J)
        W=(B(J)*A2(J)+E(J)*A1(J))
        AZ(K)=W/V
 100    CONTINUE
        WRITE(6,66)
 66     FORMAT(/,5X,'REAL',8X,'IMAGIN',8X,'MAG1',9X,'REAL',9X,'IMAGIN',3X
      +,'MAG2',8X,'ZERO',8X,'FEEDBACK')
        DO 500 I=1,50
        WRITE(6,29)XX1(I),YY1(I),YM1(I),XX2(I),YY2(I),YM2(I),AZ(I),F2(I)
 29     FORMAT(8F13.6)
 500    CONTINUE
 300    CONTINUE
        STOP
        END
```

A1= 0.950000 A2= 0.200000 B= 0.065000

E= 0.010000

| REAL | IMAGIN | MAG1 | REAL | IMAGIN | MAG2 | ZERO | FEEDBACK |
|---|---|---|---|---|---|---|---|
| 0.949999 | 0.000000 | 0.949999 | 0.200000 | 0.000000 | 0.200000 | 0.300000 | 0.000000 |
| 0.937034 | 0.000000 | 0.937034 | 0.197965 | 0.000000 | 0.197965 | 0.300000 | -0.200000 |
| 0.924141 | 0.000000 | 0.924141 | 0.195857 | 0.000000 | 0.195857 | 0.300000 | -0.400000 |
| 0.911325 | 0.000000 | 0.911325 | 0.193674 | 0.000000 | 0.193674 | 0.300000 | -0.600000 |
| 0.898588 | 0.000000 | 0.898588 | 0.191411 | 0.000000 | 0.191411 | 0.300000 | -0.800000 |
| 0.885932 | 0.000000 | 0.885932 | 0.189066 | 0.000000 | 0.189066 | 0.300000 | -1.000000 |
| 0.873364 | 0.000000 | 0.873364 | 0.186635 | 0.000000 | 0.186635 | 0.300000 | -1.200000 |
| 0.860887 | 0.000000 | 0.860887 | 0.184112 | 0.000000 | 0.184112 | 0.300000 | -1.400000 |
| 0.848504 | 0.000000 | 0.848504 | 0.181496 | 0.000000 | 0.181496 | 0.300000 | -1.599999 |
| 0.836218 | 0.000000 | 0.836218 | 0.178781 | 0.000000 | 0.178781 | 0.300000 | -1.799999 |
| 0.824037 | 0.000000 | 0.824037 | 0.175963 | 0.000000 | 0.175963 | 0.300000 | -1.999999 |
| 0.811962 | 0.000000 | 0.811962 | 0.173038 | 0.000000 | 0.173038 | 0.300000 | -2.200000 |
| 0.800000 | 0.000000 | 0.800000 | 0.170000 | 0.000000 | 0.170000 | 0.300000 | -2.400000 |
| 0.788154 | 0.000000 | 0.788154 | 0.166845 | 0.000000 | 0.166845 | 0.300000 | -2.599999 |
| 0.776431 | 0.000000 | 0.776431 | 0.163569 | 0.000000 | 0.163569 | 0.300000 | -2.799999 |
| 0.764834 | 0.000000 | 0.764834 | 0.160165 | 0.000000 | 0.160165 | 0.300000 | -2.999999 |
| 0.753370 | 0.000000 | 0.753370 | 0.156629 | 0.000000 | 0.156629 | 0.300000 | -3.200000 |
| 0.742043 | 0.000000 | 0.742043 | 0.152956 | 0.000000 | 0.152956 | 0.300000 | -3.400000 |
| 0.730860 | 0.000000 | 0.730860 | 0.149139 | 0.000000 | 0.149139 | 0.300000 | -3.599999 |
| 0.719825 | 0.000000 | 0.719825 | 0.145174 | 0.000000 | 0.145174 | 0.300000 | -3.799999 |
| 0.708945 | 0.000000 | 0.708945 | 0.141055 | 0.000000 | 0.141055 | 0.300000 | -3.999999 |
| 0.698224 | 0.000000 | 0.698224 | 0.136776 | 0.000000 | 0.136775 | 0.300000 | -4.199999 |
| 0.687668 | 0.000000 | 0.687668 | 0.132331 | 0.000000 | 0.132331 | 0.300000 | -4.400000 |
| 0.677283 | 0.000000 | 0.677283 | 0.127716 | 0.000000 | 0.127716 | 0.300000 | -4.599999 |
| 0.667075 | 0.000000 | 0.667075 | 0.122925 | 0.000000 | 0.122925 | 0.300000 | -4.799999 |
| 0.657048 | 0.000000 | 0.657048 | 0.117952 | 0.000000 | 0.117952 | 0.300000 | -4.999999 |
| 0.647207 | 0.000000 | 0.647207 | 0.112792 | 0.000000 | 0.112792 | 0.300000 | -5.199999 |
| 0.637558 | 0.000000 | 0.637558 | 0.107441 | 0.000000 | 0.107441 | 0.300000 | -5.400000 |
| 0.628106 | 0.000000 | 0.628106 | 0.101894 | 0.000000 | 0.101894 | 0.300000 | -5.599999 |
| 0.618854 | 0.000000 | 0.618854 | 0.096145 | 0.000000 | 0.096145 | 0.300000 | -5.799999 |
| 0.609807 | 0.000000 | 0.609807 | 0.090192 | 0.000000 | 0.090192 | 0.300000 | -5.999999 |
| 0.600968 | 0.000000 | 0.600968 | 0.084031 | 0.000000 | 0.084031 | 0.300000 | -6.199999 |
| 0.592342 | 0.000000 | 0.592342 | 0.077658 | 0.000000 | 0.077658 | 0.300000 | -6.400000 |
| 0.583930 | 0.000000 | 0.583929 | 0.071070 | 0.000000 | 0.071070 | 0.300000 | -6.599999 |
| 0.575734 | 0.000000 | 0.575734 | 0.064266 | 0.000000 | 0.064266 | 0.300000 | -6.799999 |
| 0.567757 | 0.000000 | 0.567757 | 0.057243 | 0.000000 | 0.057243 | 0.300000 | -6.999999 |
| 0.559999 | 0.000000 | 0.559999 | 0.050000 | 0.000000 | 0.050000 | 0.300000 | -7.199999 |
| 0.552463 | 0.000000 | 0.552463 | 0.042537 | 0.000000 | 0.042537 | 0.300000 | -7.399999 |
| 0.545146 | 0.000000 | 0.545146 | 0.034853 | 0.000000 | 0.034853 | 0.300000 | -7.599999 |
| 0.538050 | 0.000000 | 0.538050 | 0.026949 | 0.000000 | 0.026949 | 0.300000 | -7.799999 |
| 0.531173 | 0.000000 | 0.531173 | 0.018826 | 0.000000 | 0.018826 | 0.300000 | -7.999999 |
| 0.524514 | 0.000000 | 0.524514 | 0.010486 | 0.000000 | 0.010486 | 0.300000 | -8.199999 |
| 0.518069 | 0.000000 | 0.518069 | 0.001930 | 0.000000 | 0.001930 | 0.300000 | -8.399999 |
| 0.511837 | 0.000000 | 0.511837 | -0.006838 | 0.000000 | 0.006838 | 0.300000 | -8.599999 |
| 0.505816 | 0.000000 | 0.505815 | -0.015816 | 0.000000 | 0.015816 | 0.300000 | -8.799999 |
| 0.500000 | 0.000000 | 0.500000 | -0.025000 | 0.000000 | 0.025000 | 0.300000 | -8.999999 |
| 0.494386 | 0.000000 | 0.494386 | -0.034386 | 0.000000 | 0.034386 | 0.300000 | -9.199999 |
| 0.488970 | 0.000000 | 0.488970 | -0.043970 | 0.000000 | 0.043970 | 0.300000 | -9.399999 |
| 0.483747 | 0.000000 | 0.483747 | -0.053747 | 0.000000 | 0.053747 | 0.300000 | -9.599999 |
| 0.478712 | 0.000000 | 0.478712 | -0.063713 | 0.000000 | 0.063712 | 0.300000 | -9.799999 |

A1= 0.900000 A2= 0.100000 B= 0.100000

E= -0.010000

| REAL | IMAGIN | MAG1 | REAL | IMAGIN | MAG2 | ZERO | FEEDBACK |
|---|---|---|---|---|---|---|---|
| 0.900000 | 0.000000 | 0.900000 | 0.100000 | 0.000000 | 0.100000 | 0.011111 | 0.000000 |
| 0.879949 | 0.000000 | 0.879949 | 0.102051 | 0.000000 | 0.102051 | 0.011111 | -0.200000 |
| 0.859788 | 0.000000 | 0.859788 | 0.104212 | 0.000000 | 0.104212 | 0.011111 | -0.400000 |
| 0.839509 | 0.000000 | 0.839509 | 0.106491 | 0.000000 | 0.106491 | 0.011111 | -0.600000 |
| 0.819100 | 0.000000 | 0.819100 | 0.108900 | 0.000000 | 0.108900 | 0.011111 | -0.800000 |
| 0.798548 | 0.000000 | 0.798548 | 0.111452 | 0.000000 | 0.111452 | 0.011111 | -1.000000 |
| 0.777837 | 0.000000 | 0.777837 | 0.114163 | 0.000000 | 0.114163 | 0.011111 | -1.200000 |
| 0.756951 | 0.000000 | 0.756951 | 0.117048 | 0.000000 | 0.117048 | 0.011111 | -1.400000 |
| 0.735870 | 0.000000 | 0.735870 | 0.120130 | 0.000000 | 0.120130 | 0.011111 | -1.599999 |
| 0.714569 | 0.000000 | 0.714569 | 0.123431 | 0.000000 | 0.123431 | 0.011111 | -1.799999 |
| 0.693019 | 0.000000 | 0.693019 | 0.126980 | 0.000000 | 0.126980 | 0.011111 | -1.999999 |
| 0.671187 | 0.000000 | 0.671187 | 0.130813 | 0.000000 | 0.130813 | 0.011111 | -2.200000 |
| 0.649029 | 0.000000 | 0.649029 | 0.134971 | 0.000000 | 0.134971 | 0.011111 | -2.400000 |
| 0.626493 | 0.000000 | 0.626493 | 0.139507 | 0.000000 | 0.139506 | 0.011111 | -2.599999 |
| 0.603513 | 0.000000 | 0.603513 | 0.144487 | 0.000000 | 0.144487 | 0.011111 | -2.799999 |
| 0.580000 | 0.000000 | 0.580000 | 0.150000 | 0.000000 | 0.150000 | 0.011111 | -2.999999 |
| 0.555840 | 0.000000 | 0.555840 | 0.156160 | 0.000000 | 0.156160 | 0.011111 | -3.200000 |
| 0.530872 | 0.000000 | 0.530872 | 0.163128 | 0.000000 | 0.163127 | 0.011111 | -3.400000 |
| 0.504866 | 0.000000 | 0.504866 | 0.171134 | 0.000000 | 0.171134 | 0.011111 | -3.599999 |
| 0.477463 | 0.000000 | 0.477462 | 0.180538 | 0.000000 | 0.180538 | 0.011111 | -3.799999 |
| 0.448063 | 0.000000 | 0.448063 | 0.191937 | 0.000000 | 0.191937 | 0.011111 | -3.999999 |
| 0.415504 | 0.000000 | 0.415504 | 0.206496 | 0.000000 | 0.206496 | 0.011111 | -4.199999 |
| 0.376860 | 0.000000 | 0.376860 | 0.227140 | 0.000000 | 0.227140 | 0.011111 | -4.400000 |
| 0.314193 | 0.000000 | 0.314192 | 0.271808 | 0.000000 | 0.271806 | 0.011111 | -4.599999 |
| 0.284000 | 0.067408 | 0.291890 | 0.284000 | -0.067408 | 0.291890 | 0.011111 | -4.799999 |
| 0.275000 | 0.096824 | 0.291547 | 0.275000 | -0.096824 | 0.291547 | 0.011111 | -4.999999 |
| 0.266000 | 0.118507 | 0.291204 | 0.266000 | -0.118507 | 0.291204 | 0.011111 | -5.199999 |
| 0.257000 | 0.136202 | 0.290861 | 0.257000 | -0.136202 | 0.290861 | 0.011111 | -5.400000 |
| 0.248000 | 0.151314 | 0.290517 | 0.248000 | -0.151314 | 0.290517 | 0.011111 | -5.599999 |
| 0.239000 | 0.164557 | 0.290172 | 0.239000 | -0.164557 | 0.290172 | 0.011111 | -5.799999 |
| 0.230000 | 0.176352 | 0.289827 | 0.230000 | -0.176352 | 0.289827 | 0.011111 | -5.999999 |
| 0.221000 | 0.186973 | 0.289482 | 0.221000 | -0.186973 | 0.289482 | 0.011111 | -6.199999 |
| 0.212000 | 0.196611 | 0.289136 | 0.212000 | -0.196611 | 0.289136 | 0.011111 | -6.400000 |
| 0.203000 | 0.205404 | 0.288790 | 0.203000 | -0.205404 | 0.288790 | 0.011111 | -6.599999 |
| 0.194000 | 0.213457 | 0.288444 | 0.194000 | -0.213457 | 0.288444 | 0.011111 | -6.799999 |
| 0.185000 | 0.220650 | 0.288097 | 0.185000 | -0.220650 | 0.288097 | 0.011111 | -6.999999 |
| 0.176000 | 0.227649 | 0.287750 | 0.176000 | -0.227649 | 0.287750 | 0.011111 | -7.199999 |
| 0.167000 | 0.233904 | 0.287402 | 0.167000 | -0.233904 | 0.287402 | 0.011111 | -7.399999 |
| 0.158000 | 0.239658 | 0.287054 | 0.158000 | -0.239658 | 0.287054 | 0.011111 | -7.599999 |
| 0.149000 | 0.244947 | 0.286705 | 0.149000 | -0.244947 | 0.286705 | 0.011111 | -7.799999 |
| 0.140000 | 0.249600 | 0.286356 | 0.140000 | -0.249600 | 0.286356 | 0.011111 | -7.999999 |
| 0.131000 | 0.254242 | 0.286007 | 0.131000 | -0.254242 | 0.286007 | 0.011111 | -8.199999 |
| 0.122000 | 0.258294 | 0.285657 | 0.122000 | -0.258294 | 0.285657 | 0.011111 | -8.399999 |
| 0.113000 | 0.261975 | 0.285306 | 0.113000 | -0.261975 | 0.285306 | 0.011111 | -8.599999 |
| 0.104000 | 0.265300 | 0.284956 | 0.104000 | -0.265300 | 0.284956 | 0.011111 | -8.799999 |
| 0.095000 | 0.268281 | 0.284605 | 0.095000 | -0.268281 | 0.284605 | 0.011111 | -8.999999 |
| 0.086000 | 0.270931 | 0.284253 | 0.086000 | -0.270931 | 0.284253 | 0.011111 | -9.199999 |
| 0.077000 | 0.273260 | 0.283901 | 0.077000 | -0.273260 | 0.283901 | 0.011111 | -9.399999 |
| 0.068000 | 0.275274 | 0.283549 | 0.068000 | -0.275274 | 0.283549 | 0.011111 | -9.599999 |
| 0.059000 | 0.276982 | 0.283196 | 0.059000 | -0.276982 | 0.283196 | 0.011111 | -9.799999 |

A1= 0.900000 A2= 0.800000 F= 0.080000

E= 0.020000

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| REAL | IMAGIN | MAG1 | REAL | IMAGIN | MAG2 | ZERO | FEEDBACK |
| 0.899998 | 0.000000 | 0.899998 | 0.800002 | 0.000000 | 0.800002 | 0.819999 | 0.000000 |
| 0.884714 | 0.000000 | 0.884714 | 0.795285 | 0.000000 | 0.795285 | 0.819999 | -0.200000 |
| 0.871230 | 0.000000 | 0.871230 | 0.788770 | 0.000000 | 0.788770 | 0.819999 | -0.400000 |
| 0.859991 | 0.000000 | 0.859991 | 0.780008 | 0.000000 | 0.780009 | 0.819999 | -0.600000 |
| 0.851227 | 0.000000 | 0.851227 | 0.768773 | 0.000000 | 0.768773 | 0.819999 | -0.800000 |
| 0.844714 | 0.000000 | 0.844714 | 0.755285 | 0.000000 | 0.755285 | 0.819999 | -1.000000 |
| 0.839995 | 0.000000 | 0.839995 | 0.740004 | 0.000000 | 0.740004 | 0.819999 | -1.200000 |
| 0.836562 | 0.000000 | 0.836562 | 0.723437 | 0.000000 | 0.723437 | 0.819999 | -1.400000 |
| 0.834028 | 0.000000 | 0.834028 | 0.705972 | 0.000000 | 0.705972 | 0.819999 | -1.599999 |
| 0.832107 | 0.000000 | 0.832107 | 0.687892 | 0.000000 | 0.687892 | 0.819999 | -1.799999 |
| 0.830621 | 0.000000 | 0.830621 | 0.669379 | 0.000000 | 0.669379 | 0.819999 | -1.999999 |
| 0.829439 | 0.000000 | 0.829439 | 0.650560 | 0.000000 | 0.650560 | 0.819999 | -2.200000 |
| 0.828482 | 0.000000 | 0.828482 | 0.631516 | 0.000000 | 0.631516 | 0.819999 | -2.400000 |
| 0.827700 | 0.000000 | 0.827700 | 0.612299 | 0.000000 | 0.612299 | 0.819999 | -2.599999 |
| 0.827042 | 0.000000 | 0.827042 | 0.592957 | 0.000000 | 0.592956 | 0.819999 | -2.799999 |
| 0.826488 | 0.000000 | 0.826488 | 0.573511 | 0.000000 | 0.573511 | 0.819999 | -2.999999 |
| 0.826010 | 0.000000 | 0.826010 | 0.553989 | 0.000000 | 0.553989 | 0.819999 | -3.200000 |
| 0.825600 | 0.000000 | 0.825600 | 0.534399 | 0.000000 | 0.534399 | 0.819999 | -3.400000 |
| 0.825237 | 0.000000 | 0.825237 | 0.514761 | 0.000000 | 0.514761 | 0.819999 | -3.599999 |
| 0.824922 | 0.000000 | 0.824922 | 0.495078 | 0.000000 | 0.495078 | 0.819999 | -3.799999 |
| 0.824639 | 0.000000 | 0.824639 | 0.475360 | 0.000000 | 0.475360 | 0.819999 | -3.999999 |
| 0.824389 | 0.000000 | 0.824389 | 0.455611 | 0.000000 | 0.455610 | 0.819999 | -4.199999 |
| 0.824162 | 0.000000 | 0.824161 | 0.435837 | 0.000000 | 0.435837 | 0.819999 | -4.400000 |
| 0.823959 | 0.000000 | 0.823959 | 0.416041 | 0.000000 | 0.416041 | 0.819999 | -4.599999 |
| 0.823772 | 0.000000 | 0.823772 | 0.396227 | 0.000000 | 0.396227 | 0.819999 | -4.799999 |
| 0.823605 | 0.000000 | 0.823605 | 0.376395 | 0.000000 | 0.376395 | 0.819999 | -4.999999 |
| 0.823450 | 0.000000 | 0.823450 | 0.356550 | 0.000000 | 0.356550 | 0.819999 | -5.199999 |
| 0.823309 | 0.000000 | 0.823309 | 0.336691 | 0.000000 | 0.336691 | 0.819999 | -5.400000 |
| 0.823177 | 0.000000 | 0.823177 | 0.316822 | 0.000000 | 0.316822 | 0.819999 | -5.599999 |
| 0.823057 | 0.000000 | 0.823057 | 0.296943 | 0.000000 | 0.296943 | 0.819999 | -5.799999 |
| 0.822944 | 0.000000 | 0.822944 | 0.277055 | 0.000000 | 0.277055 | 0.819999 | -5.999999 |
| 0.822841 | 0.000000 | 0.822841 | 0.257159 | 0.000000 | 0.257159 | 0.819999 | -6.199999 |
| 0.822743 | 0.000000 | 0.822743 | 0.237256 | 0.000000 | 0.237256 | 0.819999 | -6.400000 |
| 0.822653 | 0.000000 | 0.822653 | 0.217347 | 0.000000 | 0.217346 | 0.819999 | -6.599999 |
| 0.822567 | 0.000000 | 0.822567 | 0.197432 | 0.000000 | 0.197432 | 0.819999 | -6.799999 |
| 0.822489 | 0.000000 | 0.822489 | 0.177511 | 0.000000 | 0.177511 | 0.819999 | -6.999999 |
| 0.822414 | 0.000000 | 0.822414 | 0.157586 | 0.000000 | 0.157586 | 0.819999 | -7.199999 |
| 0.822344 | 0.000000 | 0.822344 | 0.137656 | 0.000000 | 0.137656 | 0.819999 | -7.399999 |
| 0.822277 | 0.000000 | 0.822277 | 0.117723 | 0.000000 | 0.117723 | 0.819999 | -7.599999 |
| 0.822214 | 0.000000 | 0.822214 | 0.097786 | 0.000000 | 0.097786 | 0.819999 | -7.799999 |
| 0.822155 | 0.000000 | 0.822154 | 0.077845 | 0.000000 | 0.077845 | 0.819999 | -7.999999 |
| 0.822098 | 0.000000 | 0.822098 | 0.057902 | 0.000000 | 0.057902 | 0.819999 | -8.199999 |
| 0.822045 | 0.000000 | 0.822045 | 0.037955 | 0.000000 | 0.037955 | 0.819999 | -8.399999 |
| 0.821994 | 0.000000 | 0.821994 | 0.018006 | 0.000000 | 0.018006 | 0.819999 | -8.599999 |
| 0.821945 | 0.000000 | 0.821945 | -0.001945 | 0.000000 | 0.001945 | 0.819999 | -8.799999 |
| 0.821899 | 0.000000 | 0.821899 | -0.021899 | 0.000000 | 0.021899 | 0.819999 | -8.999999 |
| 0.821855 | 0.000000 | 0.821855 | -0.041855 | 0.000000 | 0.041855 | 0.819999 | -9.199999 |
| 0.821813 | 0.000000 | 0.821813 | -0.061813 | 0.000000 | 0.061813 | 0.819999 | -9.399999 |
| 0.821773 | 0.000000 | 0.821773 | -0.081773 | 0.000000 | 0.081773 | 0.819999 | -9.599999 |
| 0.821735 | 0.000000 | 0.821735 | -0.101735 | 0.000000 | 0.101735 | 0.819999 | -9.799999 |

A1= 0.900000 A2= 1.700000 C= 0.500000

L= -0.300000

| REAL | IMAGIN | MAG1 | REAL | IMAGIN | MAG2 | ZERO | FEEDBACK |
|---|---|---|---|---|---|---|---|
| 0.899997 | 0.000000 | 0.899997 | 0.700092 | 0.000000 | 0.700092 | 0.400000 | 0.000000 |
| 0.952473 | 0.000000 | 0.952478 | 0.657521 | 0.000000 | 0.657521 | 0.400000 | 0.200000 |
| 1.046604 | 0.000000 | 1.046603 | 0.631194 | 0.000000 | 0.631194 | 0.400000 | 0.400000 |
| 1.106191 | 0.000000 | 1.106191 | 0.611607 | 0.000000 | 0.611607 | 0.400000 | 0.600000 |
| 1.163547 | 0.000000 | 1.163547 | 0.596451 | 0.000000 | 0.596451 | 0.400000 | 0.800000 |
| 1.216226 | 0.000000 | 1.216225 | 0.583773 | 0.000000 | 0.583773 | 0.400000 | 1.000000 |
| 1.266985 | 0.000000 | 1.266984 | 0.573013 | 0.000000 | 0.573013 | 0.400000 | 1.200000 |
| 1.316296 | 0.000000 | 1.316295 | 0.563703 | 0.000000 | 0.563703 | 0.400000 | 1.400000 |
| 1.364473 | 0.000000 | 1.364472 | 0.555525 | 0.000000 | 0.555525 | 0.400000 | 1.599999 |
| 1.411738 | 0.000000 | 1.411737 | 0.548260 | 0.000000 | 0.548260 | 0.400000 | 1.799999 |
| 1.458256 | 0.000000 | 1.458255 | 0.541742 | 0.000000 | 0.541742 | 0.400000 | 1.999999 |
| 1.504147 | 0.000000 | 1.504146 | 0.535851 | 0.000000 | 0.535851 | 0.400000 | 2.200000 |
| 1.549507 | 0.000000 | 1.549506 | 0.530491 | 0.000000 | 0.530491 | 0.400000 | 2.400000 |
| 1.594414 | 0.000000 | 1.594413 | 0.525585 | 0.000000 | 0.525585 | 0.400000 | 2.599999 |
| 1.638926 | 0.000000 | 1.638925 | 0.521073 | 0.000000 | 0.521073 | 0.400000 | 2.799999 |
| 1.683093 | 0.000000 | 1.683092 | 0.516905 | 0.000000 | 0.516905 | 0.400000 | 2.999999 |
| 1.726957 | 0.000000 | 1.726956 | 0.513041 | 0.000000 | 0.513041 | 0.400000 | 3.200000 |
| 1.770554 | 0.000000 | 1.770553 | 0.509445 | 0.000000 | 0.509445 | 0.400000 | 3.400000 |
| 1.813910 | 0.000000 | 1.813909 | 0.506089 | 0.000000 | 0.506089 | 0.400000 | 3.599999 |
| 1.857050 | 0.000000 | 1.857049 | 0.502948 | 0.000000 | 0.502948 | 0.400000 | 3.799999 |
| 1.899998 | 0.000000 | 1.899997 | 0.500000 | 0.000000 | 0.500000 | 0.400000 | 3.999999 |
| 1.942770 | 0.000000 | 1.942769 | 0.497229 | 0.000000 | 0.497229 | 0.400000 | 4.199999 |
| 1.985384 | 0.000000 | 1.985383 | 0.494615 | 0.000000 | 0.494615 | 0.400000 | 4.400000 |
| 2.027852 | 0.000000 | 2.027851 | 0.492146 | 0.000000 | 0.492146 | 0.400000 | 4.599999 |
| 2.070188 | 0.000000 | 2.070187 | 0.489810 | 0.000000 | 0.489810 | 0.400000 | 4.799999 |
| 2.112401 | 0.000000 | 2.112400 | 0.487597 | 0.000000 | 0.487597 | 0.400000 | 4.999999 |
| 2.154504 | 0.000000 | 2.154503 | 0.485495 | 0.000000 | 0.485495 | 0.400000 | 5.199999 |
| 2.196503 | 0.000000 | 2.196502 | 0.483496 | 0.000000 | 0.483496 | 0.400000 | 5.400000 |
| 2.238405 | 0.000000 | 2.238404 | 0.481593 | 0.000000 | 0.481593 | 0.400000 | 5.599999 |
| 2.280220 | 0.000000 | 2.280219 | 0.479778 | 0.000000 | 0.479778 | 0.400000 | 5.799999 |
| 2.321952 | 0.000000 | 2.321951 | 0.478046 | 0.000000 | 0.478046 | 0.400000 | 5.999999 |
| 2.363607 | 0.000000 | 2.363606 | 0.476391 | 0.000000 | 0.476391 | 0.400000 | 6.199999 |
| 2.405193 | 0.000000 | 2.405192 | 0.474806 | 0.000000 | 0.474806 | 0.400000 | 6.400000 |
| 2.446711 | 0.000000 | 2.446710 | 0.473288 | 0.000000 | 0.473288 | 0.400000 | 6.599999 |
| 2.488166 | 0.000000 | 2.488165 | 0.471833 | 0.000000 | 0.471833 | 0.400000 | 6.799999 |
| 2.529562 | 0.000000 | 2.529561 | 0.470437 | 0.000000 | 0.470437 | 0.400000 | 6.999999 |
| 2.570903 | 0.000000 | 2.570902 | 0.469096 | 0.000000 | 0.469096 | 0.400000 | 7.199999 |
| 2.612192 | 0.000000 | 2.612191 | 0.467806 | 0.000000 | 0.467806 | 0.400000 | 7.399999 |
| 2.653434 | 0.000000 | 2.653433 | 0.466565 | 0.000000 | 0.466565 | 0.400000 | 7.599999 |
| 2.694629 | 0.000000 | 2.694628 | 0.465370 | 0.000000 | 0.465370 | 0.400000 | 7.799999 |
| 2.735780 | 0.000000 | 2.735779 | 0.464219 | 0.000000 | 0.464219 | 0.400000 | 7.999999 |
| 2.776891 | 0.000000 | 2.776890 | 0.463108 | 0.000000 | 0.463108 | 0.400000 | 8.199999 |
| 2.817963 | 0.000000 | 2.817962 | 0.462036 | 0.000000 | 0.462036 | 0.400000 | 8.399999 |
| 2.858998 | 0.000000 | 2.858997 | 0.461000 | 0.000000 | 0.461000 | 0.400000 | 8.599999 |
| 2.899998 | 0.000000 | 2.899997 | 0.460001 | 0.000000 | 0.460000 | 0.400000 | 8.799999 |
| 2.940966 | 0.000000 | 2.940965 | 0.459033 | 0.000000 | 0.459033 | 0.400000 | 8.999999 |
| 2.981901 | 0.000000 | 2.981900 | 0.458097 | 0.000000 | 0.458097 | 0.400000 | 9.199999 |
| 3.022808 | 0.000000 | 3.022807 | 0.457191 | 0.000000 | 0.457191 | 0.400000 | 9.399999 |
| 3.063685 | 0.000000 | 3.063684 | 0.456312 | 0.000000 | 0.456312 | 0.400000 | 9.599999 |
| 3.104535 | 0.000000 | 3.104534 | 0.455463 | 0.000000 | 0.455463 | 0.400000 | 9.799999 |

```
$JOB          WATFIV,KP=29,PAGE=100,TIME=100
C       RTL2:
C       ROOTLOCUS INTERPRETATION OF THE UNADAPTED PLANT.
C       T.F.:Y(Z)/R(Z)=((B+E)*Z-(B*A2+E*A1))/(Z**2-(A1+A2+F*E)+A1*A2+B*A2
C       F)
        DIMENSION E(4),XX1(500),XX2(500),YY1(500),YY2(500),AZ(500),F2(500)
        DIMENSION A1(4),A2(4),B(4)
        COMPLEX GC,FC,ZC(2)
        DATA E/0.01,-0.01,0.02,-0.3/
        DATA A1/0.95,0.9,0.9,0.9/
        DATA A2/0.2,0.1,0.8,0.7/
        DATA B/0.005,0.1,0.08,0.5/
        DO 300 J=1,4
        WRITE(6,200)A1(J),A2(J),B(J)
 200    FORMAT(1X,'A1=',F13.6,5X,'A2=',F13.6,5X,'B=',F13.6)
        WRITE(6,400)E(J)
 400    FORMAT(//,'                                          E=',F13.6)
        DO 100 K=1,50
        IF(J.EQ.4) GO TO 17
        F2(K)=-0.2*(K-1)
        GO TO 1
 17     F2(K)=0.2*(K-1)
 1      G=(A1(J)+A2(J)+F2(K)*E(J))
        F=(A1(J)*A2(J)+F2(K)*B(J)*A2(J))
        GC=CMPLX(G,0.0)
        FC=CMPLX(F,0.0)
        ZC(1)=(GC+CSQRT(GC**2-4.0*FC))/2.0
        ZC(2)=(GC-CSQRT(GC**2-4.0*FC))/2.0
        ZR1=REAL(ZC(1))
        ZR2=REAL(ZC(2))
        ZI2=AIMAG(ZC(2))
        ZI1=AIMAG(ZC(1))
        XX1(K)=ZR1
        XX2(K)=ZR2
        YY1(K)=ZI1
        YY2(K)=ZI2
        V=B(J)+E(J)
        W=(B(J)*A2(J)+E(J)*A1(J))
        AZ(K)=W/V
 100    CONTINUE
        WRITE(6,66)
 66     FORMAT(/,5X,'REAL',5X,'IMAGIN',8X,'REAL',5X,'IMAGIN',8X,'ZERO',5X
     +  ,'FEEDBACK')
        DO 500 I=1,50
        WRITE(6,29)XX1(I),YY1(I),XX2(I),YY2(I),AZ(I),F2(I)
 29     FORMAT(6F13.6)
 500    CONTINUE
 300    CONTINUE
        STOP
        END
```

A1= [illegible] [illegible] [illegible] = [illegible]

= [illegible]

| REAL | IMAGIN | REAL | IMAGIN | ZERO | FEEDBACK |
|---|---|---|---|---|---|
| 0.949999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | 0.000000 |
| 0.936999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -0.200000 |
| 0.923999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -0.400000 |
| 0.910999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -0.600000 |
| 0.897999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -0.800000 |
| 0.884999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -1.000000 |
| 0.871999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -1.200000 |
| 0.858999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -1.400000 |
| 0.845999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -1.599999 |
| 0.832999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -1.799999 |
| 0.819999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -1.999999 |
| 0.806999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -2.200000 |
| 0.793999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -2.400000 |
| 0.781000 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -2.599999 |
| 0.768000 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -2.799999 |
| 0.755000 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -2.999999 |
| 0.742000 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -3.200000 |
| 0.729000 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -3.400000 |
| 0.716000 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -3.599999 |
| 0.703000 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -3.799999 |
| 0.690000 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -3.999999 |
| 0.677000 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -4.199999 |
| 0.664000 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -4.400000 |
| 0.651000 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -4.599999 |
| 0.638000 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -4.799999 |
| 0.625000 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -4.999999 |
| 0.612000 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -5.199999 |
| 0.599000 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -5.400000 |
| 0.585999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -5.599999 |
| 0.572999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -5.799999 |
| 0.559999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -5.999999 |
| 0.546999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -6.199999 |
| 0.533999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -6.400000 |
| 0.520999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -6.599999 |
| 0.507999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -6.799999 |
| 0.495000 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -6.999999 |
| 0.481999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -7.199999 |
| 0.468999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -7.399999 |
| 0.455999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -7.599999 |
| 0.442999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -7.799999 |
| 0.429999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -7.999999 |
| 0.416999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -8.199999 |
| 0.403999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -8.399999 |
| 0.390999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -8.599999 |
| 0.377999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -8.799999 |
| 0.364999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -8.999999 |
| 0.351999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -9.199999 |
| 0.338999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -9.399999 |
| 0.325999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -9.599999 |
| 0.312999 | 0.000000 | 0.200000 | 0.000000 | 0.300000 | -9.799999 |

A1= [illegible] A2= [illegible] = [illegible]

= -0.01[illegible]00

| REAL | IMAGIN | REAL | IMAGIN | Z AC | F BACK |
|---|---|---|---|---|---|
| 0.900000 | [illegible] | [illegible] | [illegible] | 0.011111 | 0.000000 |
| 0.880000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | -0.200000 |
| 0.860000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | -0.400000 |
| 0.840000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | -0.600000 |
| 0.820000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | -0.800000 |
| 0.800000 | [illegible] | 0.100000 | 0.000000 | 0.011111 | -1.000000 |
| 0.780000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | -1.200000 |
| 0.760000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | -1.400000 |
| 0.740000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | [illegible] |
| 0.720000 | [illegible] | [illegible] | 0.000000 | 0.011111 | -1.799999 |
| 0.700000 | [illegible] | [illegible] | [illegible] | 0.011111 | [illegible] |
| 0.680000 | [illegible] | 0.100000 | 0.000000 | 0.011111 | [illegible] |
| 0.660000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | [illegible] |
| 0.640000 | [illegible] | 0.100000 | 0.000000 | 0.011111 | [illegible] |
| 0.620000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | [illegible] |
| 0.600000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | -2.999999 |
| 0.580000 | [illegible] | 0.100000 | 0.000000 | 0.011111 | [illegible] |
| 0.560000 | [illegible] | 0.100000 | 0.000000 | 0.011111 | [illegible] |
| 0.540000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | [illegible] |
| 0.520000 | [illegible] | 0.100000 | 0.000000 | 0.011111 | [illegible] |
| 0.500000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | [illegible] |
| 0.480000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | -4.199999 |
| 0.460000 | [illegible] | 0.100000 | 0.000000 | 0.011111 | [illegible] |
| 0.440000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | -4.599999 |
| 0.420000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | -4.799999 |
| 0.400000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | -4.999999 |
| 0.380000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | [illegible] |
| 0.360000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | [illegible] |
| 0.340000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | [illegible] |
| 0.320000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | [illegible] |
| 0.300000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | [illegible] |
| 0.280000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | -6.199999 |
| 0.260000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | [illegible] |
| 0.240000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | [illegible] |
| 0.220000 | [illegible] | [illegible] | 0.000000 | 0.011111 | [illegible] |
| 0.200000 | [illegible] | [illegible] | 0.000000 | 0.011111 | [illegible] |
| 0.180000 | 0.000000 | 0.100000 | 0.000000 | 0.011111 | [illegible] |
| 0.160000 | [illegible] | [illegible] | 0.000000 | 0.011111 | -7.399999 |
| 0.140000 | 0.000000 | [illegible] | 0.000000 | 0.011111 | [illegible] |
| 0.120001 | [illegible] | [illegible] | 0.000000 | 0.011111 | [illegible] |
| 0.100043 | [illegible] | [illegible] | 0.000000 | 0.011111 | [illegible] |
| 0.100002 | [illegible] | [illegible] | 0.000000 | 0.011111 | [illegible] |
| 0.100000 | 0.000000 | [illegible] | 0.000000 | 0.011111 | [illegible] |
| 0.100001 | [illegible] | [illegible] | 0.000000 | 0.011111 | [illegible] |
| 0.100000 | 0.000000 | [illegible] | 0.000000 | 0.011111 | [illegible] |
| 0.100000 | 0.000000 | 0.000000 | 0.000000 | 0.011111 | [illegible] |
| 0.100000 | 0.000000 | [illegible] | 0.000000 | 0.011111 | [illegible] |
| 0.100000 | 0.000000 | -0.040000 | 0.000000 | 0.011111 | [illegible] |
| 0.100000 | 0.000000 | -0.060000 | 0.000000 | 0.011111 | [illegible] |
| 0.100000 | [illegible] | [illegible] | [illegible] | 0.011111 | [illegible] |

A1= [illegible] A2= [illegible] = [illegible]

= [illegible]

| REAL | IMAGIN | REAL | IMAGIN | ZERO | FEEDBACK |
|---|---|---|---|---|---|
| 0.899993 | 0.000000 | [illegible] | 0.000000 | [illegible] | 0.000000 |
| 0.883992 | [illegible] | [illegible] | [illegible] | [illegible] | -0.200000 |
| 0.867990 | [illegible] | [illegible] | [illegible] | [illegible] | -0.400000 |
| 0.851993 | 0.000000 | [illegible] | 0.000000 | [illegible] | -0.600000 |
| 0.835993 | [illegible] | [illegible] | [illegible] | [illegible] | -0.800000 |
| 0.819994 | 0.000000 | [illegible] | [illegible] | [illegible] | -1.000000 |
| 0.803826 | 0.000000 | [illegible] | [illegible] | [illegible] | -1.200000 |
| 0.799960 | [illegible] | [illegible] | [illegible] | [illegible] | -1.400000 |
| 0.799982 | [illegible] | [illegible] | [illegible] | [illegible] | -1.599999 |
| 0.799994 | 0.000000 | [illegible] | [illegible] | [illegible] | -1.799999 |
| 0.799996 | 0.000000 | [illegible] | [illegible] | [illegible] | -1.999999 |
| 0.799983 | 0.000000 | [illegible] | 0.000000 | [illegible] | -2.200000 |
| 0.799994 | [illegible] | [illegible] | [illegible] | [illegible] | -2.400000 |
| 0.799996 | [illegible] | [illegible] | [illegible] | [illegible] | -2.599999 |
| 0.799990 | 0.000000 | [illegible] | [illegible] | [illegible] | -2.799999 |
| 0.799993 | [illegible] | [illegible] | [illegible] | [illegible] | -2.999999 |
| 0.799995 | 0.000000 | [illegible] | [illegible] | [illegible] | -3.200000 |
| 0.799997 | 0.000000 | [illegible] | [illegible] | [illegible] | -3.400000 |
| 0.799995 | 0.000000 | [illegible] | [illegible] | [illegible] | -3.599999 |
| 0.799999 | 0.000000 | [illegible] | [illegible] | [illegible] | -3.799999 |
| 0.799996 | 0.000000 | [illegible] | 0.000000 | [illegible] | -3.999999 |
| 0.799997 | [illegible] | [illegible] | [illegible] | [illegible] | -4.199999 |
| 0.799993 | 0.000000 | [illegible] | [illegible] | [illegible] | -4.400000 |
| 0.799999 | 0.000000 | [illegible] | [illegible] | [illegible] | -4.599999 |
| 0.799999 | 0.000000 | [illegible] | [illegible] | [illegible] | -4.799999 |
| 0.799997 | 0.000000 | [illegible] | [illegible] | [illegible] | -4.999999 |
| 0.799997 | 0.000000 | [illegible] | 0.000000 | [illegible] | -5.199999 |
| 0.799999 | 0.000000 | [illegible] | 0.000000 | [illegible] | -5.400000 |
| 0.799999 | 0.000000 | 0.452000 | 0.000000 | [illegible] | -5.599999 |
| 0.799995 | 0.000000 | [illegible] | [illegible] | [illegible] | -5.799999 |
| 0.799998 | 0.000000 | [illegible] | [illegible] | [illegible] | -5.999999 |
| 0.799999 | 0.000000 | 0.404001 | [illegible] | [illegible] | -6.199999 |
| 0.799999 | 0.000000 | [illegible] | 0.000000 | [illegible] | -6.400000 |
| 0.800000 | 0.000000 | [illegible] | 0.000000 | [illegible] | -6.599999 |
| 0.799993 | 0.000000 | 0.356001 | 0.000000 | [illegible] | -6.799999 |
| 0.799995 | [illegible] | 0.340001 | [illegible] | [illegible] | -6.999999 |
| 0.799995 | 0.000000 | [illegible] | [illegible] | [illegible] | -7.199999 |
| 0.799999 | 0.000000 | [illegible] | [illegible] | [illegible] | -7.399999 |
| 0.799999 | [illegible] | [illegible] | 0.000000 | [illegible] | -7.599999 |
| 0.799993 | [illegible] | [illegible] | [illegible] | [illegible] | -7.799999 |
| 0.799993 | 0.000000 | [illegible] | 0.000000 | [illegible] | -7.999999 |
| 0.799999 | 0.000000 | [illegible] | 0.000000 | [illegible] | -8.199999 |
| 0.799999 | 0.000000 | [illegible] | [illegible] | [illegible] | -8.399999 |
| 0.799993 | [illegible] | [illegible] | [illegible] | [illegible] | -8.599999 |
| 0.799999 | 0.000000 | [illegible] | 0.000000 | [illegible] | -8.799999 |
| 0.799999 | 0.000000 | [illegible] | [illegible] | [illegible] | -8.999999 |
| 0.800000 | 0.000000 | [illegible] | 0.000000 | [illegible] | -9.199999 |
| 0.800000 | [illegible] | [illegible] | [illegible] | [illegible] | -9.399999 |
| 0.800000 | 0.000000 | [illegible] | [illegible] | [illegible] | -9.599999 |
| 0.800000 | 0.000000 | [illegible] | [illegible] | [illegible] | -9.799999 |

ORIGINAL PAGE IS
OF POOR QUALITY

A1= [illegible] A2= [illegible] B= [illegible]

[illegible] -0.3[illegible]

| REAL | IMAGIN | REAL | IMAGIN | ZERO | FEEDBACK |
|---|---|---|---|---|---|
| 0.899997 | 0.000000 | 0.700002 | 0.000000 | 0.400000 | 0.000000 |
| 0.999996 | 0.000000 | 0.700002 | 0.000000 | 0.400000 | 0.200000 |
| 1.099993 | 0.000000 | 0.700001 | 0.000000 | 0.400000 | 0.400000 |
| 1.199997 | 0.000000 | 0.700001 | 0.000000 | 0.400000 | 0.600000 |
| 1.299997 | 0.000000 | 0.700001 | 0.000000 | 0.400000 | 0.800000 |
| 1.399997 | 0.000000 | 0.700001 | 0.000000 | 0.400000 | 1.000000 |
| 1.499998 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 1.200000 |
| 1.599999 | 0.000000 | 0.699999 | 0.000000 | 0.400000 | 1.400000 |
| 1.699993 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 1.599999 |
| 1.799993 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 1.799999 |
| 1.899993 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 1.999999 |
| 1.999999 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 2.200000 |
| 2.099998 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 2.400000 |
| 2.199993 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 2.599999 |
| 2.299999 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 2.799999 |
| 2.399999 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 2.999999 |
| 2.499999 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 3.200000 |
| 2.599999 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 3.400000 |
| 2.699999 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 3.599999 |
| 2.799999 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 3.799999 |
| 2.899999 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 3.999999 |
| 2.999999 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 4.199999 |
| 3.099999 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 4.400000 |
| 3.199999 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 4.599999 |
| 3.299999 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 4.799999 |
| 3.399993 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 4.999999 |
| 3.499993 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 5.199999 |
| 3.599993 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 5.400000 |
| 3.699993 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 5.599999 |
| 3.799996 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 5.799999 |
| 3.899993 | 0.000000 | 0.700001 | 0.000000 | 0.400000 | 5.999999 |
| 3.999998 | 0.000000 | 0.700001 | 0.000000 | 0.400000 | 6.199999 |
| 4.099998 | 0.000000 | 0.700001 | 0.000000 | 0.400000 | 6.400000 |
| 4.199993 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 6.599999 |
| 4.299998 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 6.799999 |
| 4.399998 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 6.999999 |
| 4.499993 | 0.000000 | 0.700001 | 0.000000 | 0.400000 | 7.199999 |
| 4.599999 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 7.399999 |
| 4.699999 | 0.000000 | 0.700001 | 0.000000 | 0.400000 | 7.599999 |
| 4.799998 | 0.000000 | 0.700001 | 0.000000 | 0.400000 | 7.799999 |
| 4.899997 | 0.000000 | 0.700001 | 0.000000 | 0.400000 | 7.999999 |
| 4.999997 | 0.000000 | 0.700001 | 0.000000 | 0.400000 | 8.199999 |
| 5.099997 | 0.000000 | 0.700001 | 0.000000 | 0.400000 | 8.399999 |
| 5.199999 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 8.599999 |
| 5.299993 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 8.799999 |
| 5.399993 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 8.999999 |
| 5.499998 | 0.000000 | 0.700001 | 0.000000 | 0.400000 | 9.199999 |
| 5.599993 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 9.399999 |
| 5.699998 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 9.599999 |
| 5.799999 | 0.000000 | 0.700000 | 0.000000 | 0.400000 | 9.799999 |

```
$JOB        WATFIV,KP=29,PAGES=100,TIME=100
C       [illegible]
C       ROOTLOCUS INTERPRETATION OF THE UNADAPTED PLANT.
C       T.F.:Y(Z)/R(Z)=((B+E)*Z-(B*A2+E*A1))/(Z**2-(A1+A2+B*E)+A1*A2+A1*E
C       *F)
        DIMENSION XX1(500),XX2(500),YY1(500),YY2(500),AZ(500),F2(500)
        COMPLEX GC,FC,ZC(2)
        A1=0.9
        A2=0.8
        B=0.08
        E=0.02
        WRITE(6,200)A1,A2,B
 200    FORMAT(1X,'A1=',F13.6,2X,'A2=',F13.6,2X,'B=',F13.6)
        WRITE(6,400)E
 400    FORMAT(//,'                                E=',F13.6)
        DO 100 K=1,50
        F2(K)=-0.2*(K-1)
        G=(A1+A2+F2(K)*E)
        F=(A1*A2+A1*E*F2(K))
        GC=CMPLX(G,0.0)
        FC=CMPLX(F,0.0)
        ZC(1)=(GC+CSQRT(GC**2-4.0*FC))/2.0
        ZC(2)=(GC-CSQRT(GC**2-4.0*FC))/2.0
        ZR1=REAL(ZC(1))
        ZR2=REAL(ZC(2))
        ZI2=AIMAG(ZC(2))
        ZI1=AIMAG(ZC(1))
        XX1(K)=ZR1
        XX2(K)=ZR2
        YY2(K)=ZI2
        YY1(K)=ZI1
        V=B+E
        W=(E*A2+B*A1)
        AZ(K)=W/V
 100    CONTINUE
        WRITE(6,66)
 66     FORMAT(/,5X,'REAL',6X,'IMAGIN',6X,'REAL',6X,'IMAGIN',6X,'ZERO',6X
     +  ,'FEEDBACK')
        DO 500 I=1,50
        WRITE(6,29)XX1(I),YY1(I),XX2(I),YY2(I),AZ(I),F2(I)
 29     FORMAT(6F13.6)
 500    CONTINUE
        STOP
        END
```

A1=

| REAL | IMAGIN | REAL | IMAGIN | ZERO | FEEDBACK |
|---|---|---|---|---|---|
| 0.899996 | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| 0.899992 | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| 0.899996 | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| 0.899997 | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| 0.899994 | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| 0.899994 | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | 0.776001 | [illegible] | 0.819999 | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| 0.900000 | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| 0.899994 | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| 0.899996 | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | 0.819999 | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | 0.819999 | [illegible] |
| 0.899995 | [illegible] | 0.725004 | 0.000000 | [illegible] | [illegible] |
| 0.899997 | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| 0.899997 | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| 0.899997 | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| 0.899997 | [illegible] | [illegible] | 0.000000 | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| 0.899996 | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| 0.899998 | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| 0.899999 | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| 0.899996 | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | 0.819999 | [illegible] |

```
C       PLOT1:
C       ROOTLOCUS INTERPRETATION OF THE UNADAPTED PLANT.
        DIMENSION XX1(500),XX2(500),YY1(500),YY2(500),YY3(500),F2(51)
        DIMENSION XXX(51),YYY(51),XX3(500)
        COMPLEX GC,FC,ZC(2)
        YMAX=1.25
        YMIN=1.25
        A1=0.95
        A2=0.2
        B=0.065
        E=0.01
        N=50
        DO 100 K=1,51
        F2(K)=-0.2*(K-1)
        G=(A1+A2+F2(K)*(B+E))
        F=A1*A2+F2(K)*(B*A2+E*A1)
        GC=CMPLX(G,0.0)
        FC=CMPLX(F,0.0)
        ZC(1)=(GC+CSQRT(GC**2-4.0*FC))/2.0
        ZC(2)=(GC-CSQRT(GC**2-4.0*FC))/2.0
        ZR1=REAL(ZC(1))
        ZR2=REAL(ZC(2))
        ZI2=AIMAG(ZC(2))
        ZI1=AIMAG(ZC(1))
        XX1(K)=ZI1
        XX2(K)=ZI2
        YY1(K)=ZR1
        YY2(K)=ZR2
        V=B+E
        W=B*A2+E*A1
        YY3(K)=W/V
        XXX(K)=-0.2*(K-1)
        CX=XXX(1)
        CY=YY3(1)
  100   CONTINUE
        CALL PLOTS(0.0,0.0,51)
        CALL PLOT(5.0,6.0,-3)
        CALL FACTOR(0.5)
        DT=XXX(51)/10.0
        CALL AXIS (0.0,0.0,'FEEDBACK PAR.',-13,10.0,0.0,0.0,DT)
        CALL GRID (0.0,-5.0,10,1.0,10,1.0,-30584)
        YT=ABS(YMAX)
        IF(ABS(YMIN).GT.YT) YT=ABS(YMIN)
        YSTART=-YT
        DY=YT/5.0
        CALL PLOT(XXX(1)/DT,YY3(1)/DY,3)
        DO 10 I=2,N
   10   CALL PLOT(XXX(I)/DT,YY3(I)/DY,2)
        CALL AXIS (0.0,-5.0,'PLANT POLES AND ZERO',20,10.0,90.0,YSTART,DY
     *)
        CALL NEWPEN(3)
        CALL PLOT(XXX(1)/DT,YY1(1)/DY,3)
        DO 20 I=2,N
   20   CALL PLOT(XXX(I)/DT,YY1(I)/DY,2)
        CALL PLOT(XXX(1)/DT,YY2(1)/DY,3)
        DO 30 I=2,N
   30   CALL PLOT(XXX(I)/DT,YY2(I)/DY,2)
        CALL PLOT(-0.5,-6.5,3)
        CALL PLOT(-0.2,-6.5,2)
        CALL NEWPEN(1)
        CALL PLOT(-0.5,-7.0,3)
        CALL PLOT(-0.2,-7.0,2)
        CALL PLOT(0.0,0.0,3)
        CALL SYMBOL(-0.15,-6.5,0.15,'POLES',0.0,5)
        CALL SYMBOL(-0.15,-7.0,0.15,'ZERO',0.0,4)
        CALL SYMBOL(-0.5,-7.5,0.15,'PLANT: Y(K)=(A1+A2)Y(K-1)-(A1*A2)Y(K-
     *2)+(B+E)*U(K-1)-(A1*E+B*A2)*U(K-2)',0.0,71)
        CALL SYMBOL(-0.5,-8.0,0.15,'CONTROL: U(K)=F1*R(K)+F2*Y(K)',0.0,29
     *)
        CALL SYMBOL(-0.5,-8.5,0.15,'RESULTING CHAR. EQN.: Z**2-Z*(A1+A2+F
     **(B+E))+A1*A2+F*(B*A2+E*A1)',0.0,64)
        CALL SYMBOL(-0.5,-9.0,0.15,'PARAMETERS: A1=0.95    A2=0.2    B=0.0
     *65     E=0.01',0.0,49)
        CALL PLOT(18.0,0.0,-3)
        CALL PLOT(0.0,0.0,+999)
        STOP
        END
```

## Appendix B: Supporting Alebra for (3-31) - (3-43) and Simulated Check

Contents:

From (3-31)

$$\frac{Y(z)}{R(z)} = \frac{\delta z^2[(\beta_{11}z+\beta_{12})(z^2 - \alpha_{21}z - \alpha_{22}) + (\beta_{21}z + \beta_{22})(z^2 - \alpha_{11}z - \alpha_{i2})]}{\left[\begin{array}{l}(z^2 - \eta_1 z - \eta_2)(z^2 - \alpha_{11}z - \alpha_{12})(z^2 - \alpha_{21}z - \alpha_{22}) \\ \quad - (\nu_1 z + \nu_2)[(\beta_{11}z + \beta_{12})(z^2 - \alpha_{21}z - \alpha_{22}) + (\beta_{21}z + \beta_{22})(z^2 - \alpha_{11}z - \alpha_{12})]\end{array}\right]}$$

$$= \frac{b_5 z^5 + b_4 z^4 + b_3 z^3 + b_2 z^2 + b_1 z + b_0}{a_6 z^6 + a_5 z^5 + a_4 z^4 + a_3 z^3 + a_2 z^2 + a_1 z + a_0} = \frac{N(z)}{D(z)} \ . \tag{B-1}$$

Evaluating the numerator

$$\begin{aligned} N(z) = \delta z^2 \ [&\beta_{11}z^3 - \beta_{11}\alpha_{21}z^2 - \beta_{11}\alpha_{22}z + \beta_{12}z^2 - \beta_{12}\alpha_{21}z \\ &- \beta_{12}\alpha_{22} + \beta_{21}z^3 - \beta_{21}\alpha_{11}z^2 - \beta_{21}\alpha_{12}z \\ &+ \beta_{22}z^2 - \beta_{22}\alpha_{11}z - \beta_{22}\alpha_{12}] \\ = \ &[\delta(\beta_{11} + \beta_{21})]z^5 + [\delta(-\beta_{11}\alpha_{21} + \beta_{12} - \beta_{21}\alpha_{11} + \beta_{22})]z^4 \\ &+[\delta(-\beta_{11}\alpha_{22} - \beta_{12}\alpha_{21} - \beta_{21}\alpha_{12} - \beta_{22}\alpha_{11})]z^3 \\ &+[\delta(-\beta_{12}\alpha_{22} - \beta_{22}\alpha_{12})] \ z^2 \end{aligned} \tag{B-2}$$

yields (3-32) - (3-36). Define the denominator as

$$D(z) \triangleq D_1(z) - D_2(z) \tag{B-3}$$

where

$$D_1(z) = (z^2 - \eta_1 z - \eta_2)(z^2 - \alpha_{11}z - \alpha_{12})(z^2 - \alpha_{21}z - \alpha_{22}) \tag{B-4}$$

and

$$D_2(z) = (\nu_1 \ z + \nu_2) \ \frac{N(z)}{\delta z^2} \ . \tag{B-5}$$

Expanding $D_1(z)$ yields

$$D_1(z) = (z^2 - \eta_1 z - \eta_2)(z^4 - \alpha_{21}z^3 - \alpha_{22}z^2 - \alpha_{11}z^3 + \alpha_{11}\alpha_{21}z^2 + \alpha_{11}\alpha_{22}z^2 - \alpha_{12}z^2 + \alpha_{12}\alpha_{21}z + \alpha_{12}\alpha_{22})$$

$$= (z^2 - \eta_1 z - \eta_2)[z_4 - (\alpha_{21} + \alpha_{11})z^3 - (\alpha_{22} - \alpha_{11}\alpha_{21} + \alpha_{12})z^2 + (\alpha_{11}\alpha_{22} + \alpha_{12}\alpha_{21})z + \alpha_{12}\alpha_{22}]$$

$$= z^6 - (\alpha_{21} + \alpha_{11})\ z^5 - (\alpha_{22} - \alpha_{11}\alpha_{21} + \alpha_{12})z^4 + (\alpha_{11}\alpha_{22} + \alpha_{12}\alpha_{21})z^3 + \alpha_{12}\alpha_{22}z^2 - \eta_1 z^5 + \eta_1(\alpha_{21} + \alpha_{11})z^4 + \eta_1(\alpha_{22} - \alpha_{11}\alpha_{21} + \alpha_{12})z^3 - \eta_1(\alpha_{11}\alpha_{22} + \alpha_{12}\alpha_{21})z^2 - \eta_1\alpha_{12}\alpha_{22}z - \eta_2 z^4 + \eta_2(\alpha_{21} + \alpha_{11})z^3 + \eta_2(\alpha_{22} - \alpha_{11}\alpha_{21} + \alpha_{12})z^2 - \eta_2(\alpha_{11}\alpha_{22} + \alpha_{12}\alpha_{21})z - \eta_2\alpha_{12}\alpha_{22}$$

$$= z^6 + (-\alpha_{21} - \alpha_{11} - \eta_1)z^5 + (-\alpha_{22} + \alpha_{11}\alpha_{21} - \alpha_{12} + \eta_1\alpha_{21} + \eta_1\alpha_{11} - \eta_2)z^4 + (\alpha_{11}\alpha_{22} + \alpha_{12}\alpha_{21} + \eta_1\alpha_{22} - \eta_1\alpha_{11}\alpha_{21} + \eta_1\alpha_{12} + \eta_2\alpha_{21} + \eta_2\alpha_{11})z^3 + (\alpha_{12}\alpha_{22} - \eta_1\alpha_{11}\alpha_{22} - \eta_1\alpha_{12}\alpha_{21} + \eta_2\alpha_{22} - \eta_2\alpha_{11}\alpha_{21} + \eta_2\alpha_{12})z^2 + (-\eta\ \alpha_{12}\alpha_{22} - \eta_2\alpha_{11}\alpha_{22} - \eta_2\alpha_{12}\alpha_{21})z - \eta_2\alpha_{12}\alpha_{22}\ . \tag{B-6}$$

Expanding $D_2(z)$ given (B-2) yields

$$D_2(z) = (\nu_1 z + \nu_2)[(\beta_{11} + \beta_{21})z^3 + (-\beta_{11}\alpha_{21} + \beta_{12} - \beta_{21}\alpha_{11} + \beta_{22})z^2 + (-\beta_{11}\alpha_{22} - \beta_{12}\alpha_{21} - \beta_{21}\alpha_{12} - \beta_{22}\alpha_{11})z + (-\beta_{12}\alpha_{22} - \beta_{22}\alpha_{12})]$$

$$= (\nu_1\beta_{11} + \nu_1\ \beta_{21})z^4 + (-\nu_1\beta_{11}\alpha_{21} + \nu_1\beta_{12} - \nu_1\ \beta_{21}\alpha_{11} + \nu_1\ \beta_{22}$$

$$+ \nu_2\beta_{11} + \nu_2\ \beta_{21})z^3 + (-\nu_1\beta_{11}\alpha_{22} - \nu_1\beta_{12}\alpha_{21} - \nu_1\ \beta_{21}\alpha_{12}$$

$$- \nu_1\ \beta_{22}\alpha_{11} - \nu_2\beta_{11}\alpha_{21} + \nu_2\beta_{12} - \nu_2\ \beta_{21}\alpha_{11} + \nu_2\ \beta_{22})z^2$$

$$+ (-\nu_1\beta_{12}\alpha_{22} - \nu_1\ \beta_{22}\alpha_{12} - \nu_2\beta_{11}\alpha_{22} - \nu_2\beta_{12}\alpha_{21}$$

$$-\nu_2\ \beta_{21}\alpha_{12} - \nu_2\ \beta_{22}\alpha_{11})z + (-\nu_2\beta_{12}\alpha_{22} - \nu_2\ \beta_{22}\alpha_{12}). \qquad \text{(B-7)}$$

subtracting (B-7) form (B-6) as in (B-3) yields (3-37) - (3-43).

For example 5 the unit step response from zero initial conditions of $\frac{Y(z)}{R(z)}$ in (B-1) with the parameter definitions of (3-32) - (3-43) was compared with that of (3-17) using the control law of (3-25). The program listing and outputs for this comparison complete this appendix. Note that the outputs are identical, which verifies (3-32) - (3-43).

```
$JOB           WATFIV,KP=29,PAGE=100,TIME=100
      DIMENSION G1(2),F1(2),ZET(2),W(2),AL(2),G2(2),F2(2),AU(   ),YC(200
     +),YC(200),DY(200)
      DATA T/0.5/,AL/1.0,1.0/,E/1.0/,ZET/0.2,0.02/,W/0.5,5.0/  /0.5/,GM1
     +/1.687/,GM2/-0.741/,R/1.0/,Y1/0.0/,Y2/0.0/,Y3/0.0/,Y4/   /,U1/0.0
     +/,U2/0.0/,U3/0.0/,U4/0.0/,R1/0.0/,R2/0.0/,R3/0.0/,R4/0   /,R5/0.0/,
     +R6/0.0/
      DO 10 I=1,2
      G1(I)=2*EXP(-ZET(I)*W(I)*T)*COS(W(I)*T*SQRT(1-ZET(I)**2))
      IF(I.EQ.1)A11=G1(1)
      IF(I.EQ.2)A21=G1(2)
      G2(I)=-EXP(-2*ZET(I)*W(I)*T)
      IF(I.EQ.1)A12=G2(1)
      IF(I.EQ.2)A22=G2(2)
      F1(I)=(AL(I)/W(I)**2)*(1-EXP(-ZET(I)*W(I)*T)*(COS(W(I)*T*SQRT(1-ZE
     +T(I)**2))+(ZET(I)/SQRT(1-ZET(I)**2))*SIN(W(I)*T*SQRT(1-ZET(I)**2))
     +))
      IF(I.EQ.1)B11=F1(1)
      IF(I.EQ.2)B21=F1(2)
      F2(I)=(AL(I)/W(I)**2)*(EXP(-ZET(I)*W(I)*T)*(EXP(-ZET(I)*W(I)*T)-CO
     +S(W(I)*T*SQRT(1-ZET(I)**2))+(ZET(I)/SQRT(1-ZET(I)**2))*SIN(W(I)*T*
     +SQRT(1-ZET(I)**2))))
      IF(I.EQ.1)B12=F2(1)
      IF(I.EQ.2)B22=F2(2)
10    CONTINUE
      WRITE(6,15)
15    FORMAT(1X,'           A11',10X,'A12',10X,'B11',10X,'B12',10X,'A21',10
     +X,'A22',10X,'B21',10X,'B22',10X)
      WRITE(6,20)A11,A12,B11,B12,A21,A22,B21,B22
20    FORMAT(//,8F13.6)
      ET1=GM1-A11
      ANU1=(B11*A12+(B11*A11-A12+GM2)*(A11-A12*B11/B12))/(B11*A11+B12-A1
     +2*B11**2/B12)
      ET2=A11*ET1+GM2-B11*ANU1-A12
      ANU2=ET2*A12/B12
      WRITE(6,25)
25    FORMAT(1X,'           DEL',10X,'ET1',10X,'ET2',10X,'NU1',10X,'NU2')
      WRITE(6,30)D,ET1,ET2,ANU1,ANU2
30    FORMAT(//,5F13.6)
      DO 35 J=1,200
      Y=(A21+A11)*Y1+(A22-A11*A21+A12)*Y2-(A11*A22+A12*A21)*Y3-(A12*A22)
     +*Y4+(B11+E*B21)*U1-(B11*A21-B12+E*B21*A11-E*B22)*U2-(B11*A22+B12*A
     +21+E*B21*A12+E*B22*A11)*U3-(B12*A22+E*B22*A12)*U4
      U=D*R+ET1*U1+ET2*U2+ANU1*Y1+ANU2*Y2
      YC(J)=Y
      AU(J)=U
      Y4=Y3
      Y3=Y2
      Y2=Y1
      Y1=Y
      U4=U3
      U3=U2
      U2=U1
      U1=U
35    CONTINUE
      B5=D*(B11+E*B21)
      B4=D*(-B11*A21+B12-E*B21*A11+E*B22)
      B3=D*(-B11*A22-B12*A21-E*B21*A12-E*B22*A11)
      B2=D*(-B12*A22-E*B22*A12)
      B1=0.0
```

```
      B0=B1
      A6=1.0
      A5=-A21-A11-ET1
      A4=-A22+A11*A21-A12+ET1*A21+ET1*A11-ET2-ANU1*B11-ANU1*E*B21
      A3=A11*A22+A12*A21+ET1*A22-ET1*A11*A21+ET1*A12+ET2*A21+ET2*A11+ANU
     +1*B11*A21-ANU1*B12+ANU1*E*B21*A11-ANU1*E*B22-ANU2*B11-ANU2*E*B21
      A2=A12*A22-ET1*A11*A22-ET1*A12*A21+ET2*A22-ET2*A11*A21+ET2*A12+ANU
     +1*B11*A22+ANU1*B12*A21+ANU1*E*B21*A12+ANU1*E*B22*A11+ANU2*B11*A21-
     +ANU2*B12+ANU2*E*B21*A11-ANU2*E*B22
      A1=-ET1*A12*A22-ET2*A11*A22-ET2*A12*A21+ANU1*B12*A22+ANU1*E*B22*A1
     +2+ANU2*B11*A22+ANU2*B12*A21+ANU2*E*B21*A12+ANU2*E*B22*A11
      A0=-ET2*A12*A22+ANU2*B12*A22+ANU2*E*B22*A12
      Y1=0.0
      Y2=0.0
      Y3=0.0
      Y4=0.0
      Y5=0.0
      Y6=0.0
      DO 40 K=1,200
      Y=-A5*Y1-A4*Y2-A3*Y3-A2*Y4-A1*Y5-A0*Y6+B5*R1+B4*R2+B3*R3+B2*R4+B1*
     +R5+B0*R6
      YO(K)=Y
      Y6=Y5
      Y5=Y4
      Y4=Y3
      Y3=Y2
      Y2=Y1
      Y1=Y
      R6=R5
      R5=R4
      R4=R3
      R3=R2
      R2=R1
      R1=R
   40 CONTINUE
      WRITE(6,60)
   60 FORMAT(1X,'YC: CONTROLLED SYS',5X,'YO: OVERALL T.F',12X,'(YC-YO)**
     +2')
      DO 50 L=1,200
      DY(L)=(YC(L)-YO(L))**2
      WRITE(6,55)YC(L),YO(L),DY(L)
   55 FORMAT(1X,F13.6,10X,F13.6,10X,F13.6)
   50 CONTINUE
      STOP
      END
```

| A11 | A12 | B11 | B12 | A21 |
|---|---|---|---|---|
| 1.845669 | -0.904637 | 0.120314 | 0.116359 | -1.523571 |

| A22 | B21 | B22 |
|---|---|---|
| -0.904637 | 0.070016 | 0.067121 |

| DEL | ET1 | ET2 | NU1 | NU2 |
|---|---|---|---|---|
| 0.500000 | -0.158669 | -0.071517 | -0.477264 | 0.556613 |

| YC: CONTROLLED SYS | YO: OVERALL T.F | (YC-YO)**2 |
|---|---|---|
| 0.000000 | 0.000000 | 0.000000 |
| 0.095165 | 0.095165 | 0.000000 |
| 0.229498 | 0.229498 | 0.000000 |
| 0.449052 | 0.449052 | 0.000000 |
| 0.731993 | 0.731993 | 0.000000 |
| 0.972126 | 0.972126 | 0.000000 |
| 1.269010 | 1.269011 | 0.000000 |
| 1.509712 | 1.509711 | 0.000000 |
| 1.724968 | 1.724961 | 0.000000 |
| 1.940312 | 1.940311 | 0.000000 |
| 2.073265 | 2.073263 | 0.000000 |
| 2.213954 | 2.213952 | 0.000000 |
| 2.305182 | 2.305181 | 0.000000 |
| 2.356454 | 2.356450 | 0.000000 |
| 2.415576 | 2.415572 | 0.000000 |
| 2.417261 | 2.417256 | 0.000000 |
| 2.428103 | 2.428098 | 0.000000 |
| 2.423304 | 2.423298 | 0.000000 |
| 2.392859 | 2.392852 | 0.000000 |
| 2.385548 | 2.385542 | 0.000000 |
| 2.349551 | 2.349545 | 0.000000 |
| 2.325586 | 2.325580 | 0.000000 |
| 2.307442 | 2.307438 | 0.000000 |
| 2.273036 | 2.273030 | 0.000000 |
| 2.263845 | 2.263839 | 0.000000 |
| 2.241444 | 2.241438 | 0.000000 |
| 2.226110 | 2.226102 | 0.000000 |
| 2.222950 | 2.222944 | 0.000000 |
| 2.206086 | 2.206080 | 0.000000 |
| 2.207929 | 2.207924 | 0.000000 |
| 2.203193 | 2.203189 | 0.000000 |
| 2.198542 | 2.198537 | 0.000000 |
| 2.205502 | 2.205499 | 0.000000 |
| 2.199992 | 2.199988 | 0.000000 |
| 2.205641 | 2.205638 | 0.000000 |
| 2.208404 | 2.208402 | 0.000000 |
| 2.206746 | 2.206744 | 0.000000 |
| 2.214703 | 2.214701 | 0.000000 |
| 2.212575 | 2.212573 | 0.000000 |
| 2.216134 | 2.216133 | 0.000000 |
| 2.219666 | 2.219666 | 0.000000 |
| 2.217241 | 2.217240 | 0.000000 |
| 2.222535 | 2.222535 | 0.000000 |
| 2.220882 | 2.220881 | 0.000000 |
| 2.221516 | 2.221515 | 0.000000 |
| 2.224154 | 2.224151 | 0.000000 |
| 2.221069 | 2.221067 | 0.000000 |
| 2.223914 | 2.223912 | 0.000000 |
| 2.222715 | 2.222713 | 0.000000 |
| 2.221739 | 2.221736 | 0.000000 |
| 2.223776 | 2.223774 | 0.000000 |
| 2.221040 | 2.221037 | 0.000000 |
| 2.222480 | 2.222478 | 0.000000 |
| 2.222060 | 2.222057 | 0.000000 |
| 2.220642 | 2.220639 | 0.000000 |
| 2.222396 | 2.222394 | 0.000000 |
| 2.220490 | 2.220488 | 0.000000 |
| ⋮ | ⋮ | ⋮ |
| 2.220990 | 2.220982 | 0.000000 |
| 2.220992 | 2.220979 | 0.000000 |
| 2.220990 | 2.220983 | 0.000000 |
| 2.220991 | 2.220979 | 0.000000 |
| 2.220990 | 2.220981 | 0.000000 |
| 2.220990 | 2.220982 | 0.000000 |
| 2.220989 | 2.220979 | 0.000000 |

## Appendix C: Factorization of $\frac{Y}{R}$ for Example 5

Contents:

```
C     THIS PROGRAM DETERMINES PLANT PAR. AND SING.
      DIMENSION G1(2),F1(2),ZET(2),W(2),AL(2),G2(2),F2(2)
      COMPLEX Z(4),Z1(3)
      DATA T/0.5/,AL/1.0,1.0/,ZET/0.2,0.2/,W/0.5,2.0/,ZETA/0.7/
      DO 10 I=1,2
      G1(I)=2*EXP(-ZET(I)*W(I)*T)*COS(W(I)*T*SQRT(1-ZET(I)**2))
      IF(I.EQ.1)A11=G1(1)
      IF(I.EQ.2)A21=G1(2)
      G2(I)=-EXP(-2*ZET(I)*W(I)*T)
      IF(I.EQ.1)A12=G2(1)
      IF(I.EQ.2)A22=G2(2)
      F1(I)=(AL(I)/W(I)**2)*(1-EXP(-ZET(I)*W(I)*T)*(COS(W(I)*T*SQRT(1-Z
     +ET(I)**2))+(ZET(I)/SQRT(1-ZET(I)**2))*SIN(W(I)*T*SQRT(1-ZET(I)**2))
     +))
      IF(I.EQ.1)B11=F1(1)
      IF(I.EQ.2)B21=F1(2)
      F2(I)=(AL(I)/W(I)**2)*(EXP(-ZET(I)*W(I)*T)*(EXP(-ZET(I)*W(I)*T)-CO
     +S(W(I)*T*SQRT(1-ZET(I)**2))+(ZET(I)/SQRT(1-ZET(I)**2))*SIN(W(I)*T*
     +SQRT(1-ZET(I)**2))))
      IF(I.EQ.1)B12=F2(1)
      IF(I.EQ.2)B22=F2(2)
 10   CONTINUE
      GM1=2*EXP(-ZETA*W(1)*T)*COS(W(1)*T*SQRT(1.0-ZETA**2))
      GM2=-EXP(-2*ZETA*W(1)*T)
      B3=B11+B21
      B2=-B11*A21+B12-B21*A11+B22
      B1=-B11*A22-B12*A21-B21*A12-B22*A11
      B0=-B12*A22-B22*A12
      A4=1.0
      A3=-A21-A11
      A2=-A22+A11*A21-A12
      A1=A11*A22+A12*A21
      A0=A12*A22
      WRITE(6,60)B3,B2,B1,B0
 60   FORMAT(1X,'B3=',F9.6,2X,'B2=',F9.6,2X,'B1=',F9.6,2X,'B0=',F9.6,2X)
      WRITE(6,70)A4,A3,A2,A1,A0
 70   FORMAT(1X,'A4=',F9.6,2X,'A3=',F9.6,2X,'A2=',F9.6,2X,'A1=',F9.6,2X,
     +'A0=',F9.6,2X)
      ET1=GM1-A11
      ANU1=(ET1*A12+(ET1*A11-A12+GM2)*(A11-A12*ET1/ET2))/(ET1*A11+ET2-A1
     +2*ET1**2/ET2)
      ET2=A11*ET1+GM2-ET1*ANU1-A12
      ANU2=ET2*A12/ET1
      WRITE(6,50)ET1,ET2,ANU1,ANU2,GM1,GM2
 50   FORMAT(1X,'ET1=',F9.6,2X,'ET2=',F9.6,2X,'NU1=',F9.6,2X,'NU2=',F9.6
     +,2X,'GM1=',F9.6,2X,'GM2=',F9.6,2X)
      WRITE(6,80)
 80   FORMAT(1X,'PLANT POLES AND ZEROS')
      WRITE(6,85)
 85   FORMAT(//,'              POLES                          ZEROS')
      WRITE(6,45)
 45   FORMAT(/,'             REAL',9X,'IMAG',9X,'REAL',7X,'IMAG')
      CALL [illegible](Z,A4,A3,A2,A1,A0)
      CALL [illegible](Z1,[illegible])
      DO 66 I =1,4
      ZR=REAL(Z(I))
      ZI=AIMAG(Z(I))
      Z1I=AIMAG(Z1(I))
      ZR1=REAL(Z1(I))
      IF(I.EQ.4)GO TO 15
      WRITE(6,65)ZR,ZI,ZR1,Z1I
 65   FORMAT(1X,4F15.6)
      GO TO 66
 15   WRITE(6,20)ZR,ZI
 20   FORMAT(1X,2F15.6)
 66   CONTINUE
      STOP
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE [illegible] (Z,A4,A3,A2,A1,A0)
      INTEGER NDEG,IER
      REAL P(5)
      COMPLEX Z(4)
      NDEG=4
      P(1)=A4
      P(2)=A3
      P(3)=A2
      P(4)=A1
      P(5)=A0
      CALL ZRPOLY (P,NDEG,Z,IER)
      RETURN
      END


      SUBROUTINE [illegible] (Z1,C3,C2,C1,C0)
      INTEGER NDEG,IER
      REAL C(4)
      COMPLEX Z1(3)
      NDEG=3
      C(1)=C3
      C(2)=C2
      C(3)=C1
      C(4)=C0
      CALL ZRPOLY (C,NDEG,Z1,IER)
      RETURN
      END
```

```
C   IMSL ROUTINE NAME   - ZRPOLY
C
C-----------------------------------------------------------------------
C
C   COMPUTER            - IBM/DOUBLE
C
C   LATEST REVISION     - JANUARY 1, 1978
C
C   PURPOSE             - ZEROS OF A POLYNOMIAL WITH REAL
C                           COEFFICIENTS (JENKINS-TRAUB)
C
C   USAGE               - CALL ZRPOLY (A,NDEG,Z,IER)
C
C   ARGUMENTS    A      - INPUT REAL VECTOR OF LENGTH NDEG+1
C                           CONTAINING THE COEFFICIENTS IN ORDER OF
C                           DECREASING POWERS OF THE VARIABLE.
C                NDEG   - INPUT INTEGER DEGREE OF POLYNOMIAL.
C                           NDEG MUST BE GREATER THAN 0 AND LESS
C                           THAN 101.
C                Z      - OUTPUT COMPLEX VECTOR OF LENGTH NDEG
C                           CONTAINING THE COMPUTED ROOTS OF THE
C                           POLYNOMIAL.
C                         NOTE - THE ROUTINE TREATS Z AS A REAL VECTOR
C                           OF LENGTH 2*NDEG. AN APPROPRIATE
C                           EQUIVALENCE STATEMENT MAY BE REQUIRED.
C                           SEE DOCUMENT EXAMPLE.
C                IER    - ERROR PARAMETER. (OUTPUT)
C                         TERMINAL ERROR
C                           IER=129, INDICATES THAT THE DEGREE OF THE
C                             POLYNOMIAL IS GREATER THAN 100 OR LESS
C                             THAN 1.
C                           IER=130, INDICATES THAT THE LEADING
C                             COEFFICIENT IS ZERO.
C                           IER=131, INDICATES THAT ZRPOLY FOUND FEWER
C                             THAN NDEG ZEROS. IF ONLY M ZEROS ARE
C                             FOUND, Z(J),J=M+1,...,NDEG ARE SET TO
C                             POSITIVE MACHINE INFINITY.
C
C   PRECISION/HARDWARE  - SINGLE AND DOUBLE/H32
C                       - SINGLE/H36,H48,H60
C
C   REQD. IMSL ROUTINES - UERTST,UGETIO,ZRPQLB,ZRPQLC,ZRPQLD,ZRPQLE,
C                           ZRPQLF,ZRPQLG,ZRPQLH,ZRPQLI
C
C   NOTATION            - INFORMATION ON SPECIAL NOTATION AND
C                           CONVENTIONS IS AVAILABLE IN THE MANUAL
C                           INTRODUCTION OR THROUGH IMSL ROUTINE UHELP
C
C   COPYRIGHT           - 1978 BY IMSL, INC. ALL RIGHTS RESERVED.
C
C   WARRANTY            - IMSL WARRANTS ONLY THAT IMSL TESTING HAS BEEN
C                           APPLIED TO THIS CODE. NO OTHER WARRANTY,
C                           EXPRESSED OR IMPLIED, IS APPLICABLE.
C
C-----------------------------------------------------------------------
C
      SUBROUTINE ZRPOLY (A,NDEG,Z,IER)
C                                  SPECIFICATIONS FOR ARGUMENTS
      INTEGER            NDEG,IER
      DOUBLE PRECISION   A(1),Z(1)
C                                  SPECIFICATIONS FOR LOCAL VARIABLES
      INTEGER            N,NN,J,JJ,I,NM1,ICNT,N2,L,NZ,NPI
      REAL               ETA,RMRE,RINFP,REPSP,RADIX,RLO,XX,YY,SINR,
     1                   COSR,RMAX,RMIN,X,SC,XM,FF,DX,DF,BND,XXX,ARE
      REAL               PT(101)
      DOUBLE PRECISION   TEMP(101),P(101),QP(101),RK(101),QK(101),
     1                   SVK(101)
      DOUBLE PRECISION   SR,SI,U,V,RA,RB,C,D,A1,A2,A3,
     1                   A6,A7,E,F,G,H,SZR,SZI,RLZR,RLZI,
     2                   T,AA,BB,CC,FACTOR,REPSR1,ZERO,ONE,FN
      LOGICAL            ZEROK
      COMMON /ZRPQLJ/    P,QP,RK,QK,SVK,SR,SI,U,V,RA,RB,C,D,A1,A2,A3,A6,
     1                   A7,E,F,G,H,SZR,SZI,RLZR,RLZI,ETA,ARE,RMRE,N,NN
```

```
C                                         THE FOLLOWING STATEMENTS SET MACHINE
C                                            CONSTANTS USED IN VARIOUS PARTS OF
C                                            THE PROGRAM. THE MEANING OF THE
C                                            FOUR CONSTANTS ARE - REPSR1 THE
C                                            MAXIMUM RELATIVE REPRESENTATION
C                                            ERROR WHICH CAN BE DESCRIBED AS
C                                            THE SMALLEST POSITIVE FLOATING
C                                            POINT NUMBER SUCH THAT 1.+REPSR1 IS
C                                            GREATER THAN 1
C                                         RINFP THE LARGEST FLOATING-POINT
C                                            NUMBER
C                                         REPSP THE SMALLEST POSITIVE
C                                            FLOATING-POINT NUMBER IF THE
C                                            EXPONENT RANGE DIFFERS IN SINGLE
C                                            AND DOUBLE PRECISION THEN REPSP
C                                            AND RINFP SHOULD INDICATE THE
C                                            SMALLER RANGE
C                                         RADIX THE BASE OF THE FLOATING-POINT
C                                            NUMBER SYSTEM USED
      DATA                     RINFP/Z7FFFFFFF/
      DATA                     REPSP/Z00100000/
      DATA                     RADIX/16.0/
      DATA                     REPSR1/Z3410000000000000/
      DATA                     ZERO/0.0D0/,ONE/1.0D0/
C                                         ZRPOLY USES SINGLE PRECISION
C                                            CALCULATIONS FOR SCALING, BOUNDS
C                                            AND ERROR CALCULATIONS.
C                                         FIRST EXECUTABLE STATEMENT
      IER = 0
      IF (NDEG .GT. 100 .OR. NDEG .LT. 1) GO TO 165
      ETA = REPSR1
      ARE = ETA
      RMRE = ETA
      RLO = REPSP/ETA
C                                         INITIALIZATION OF CONSTANTS FOR
C                                            SHIFT ROTATION
      XX = .7071068
      YY = -XX
      SINR = .9975641
      COSR = -.06975647
      N = NDEG
      NN = N+1
C                                         ALGORITHM FAILS IF THE LEADING
C                                            COEFFICIENT IS ZERO.
      IF (A(1).NE.ZERO) GO TO 5
      IER = 130
      GO TO 9000
C                                         REMOVE THE ZEROS AT THE ORIGIN IF
C                                            ANY
    5 IF (A(NN).NE.ZERO) GO TO 10
      J = NDEG-N+1
      JJ = J+NDEG
      Z(J) = ZERO
      Z(JJ) = ZERO
      NN = NN-1
      N = N-1
      IF (NN.EQ.1) GO TO 9005
      GO TO 5
C                                         MAKE A COPY OF THE COEFFICIENTS
   10 DO 15 I=1,NN
         P(I) = A(I)
   15 CONTINUE
C                                         START THE ALGORITHM FOR ONE ZERO
   20 IF (N.GT.2) GO TO 30
      IF (N.LT.1) GO TO 9005
C                                         CALCULATE THE FINAL ZERO OR PAIR OF
C                                            ZEROS
      IF (N.EQ.2) GO TO 25
      Z(NDEG) = -P(2)/P(1)
      Z(NDEG+NDEG) = ZERO
      GO TO 145
   25 CALL ZRPQLI (P(1),P(2),P(3),Z(NDEG-1),Z(NDEG+NDEG-1),Z(NDEG),
     1   Z(NDEG+NDEG))
      GO TO 145
C                                         FIND LARGEST AND SMALLEST MODULI OF
C                                            COEFFICIENTS.
   30 RMAX = 0.
      RMIN = RINFP
      DO 35 I=1,NN
         X = ABS(SNGL(P(I)))
         IF (X.GT.RMAX) RMAX = X
         IF (X.NE.0..AND.X.LT.RMIN) RMIN = X
   35 CONTINUE
```

```
C                                           SCALE IF THERE ARE LARGE OR VERY
C                                             SMALL COEFFICIENTS COMPUTES A
C                                             SCALE FACTOR TO MULTIPLY THE
C                                             COEFFICIENTS OF THE POLYNOMIAL.
C                                             THE SCALING IS DONE TO AVOID
C                                             OVERFLOW AND TO AVOID UNDETECTED
C                                             UNDERFLOW INTERFERING WITH THE
C                                             CONVERGENCE CRITERION.
C                                           THE FACTOR IS A POWER OF THE BASE
      SC = RLO/RMIN
      IF (SC.GT.1.0) GO TO 40
      IF (RMAX.LT.10.) GO TO 55
      IF (SC.EQ.0.) SC = REPSP*RADIX*RADIX
      GO TO 45
   40 IF (RINFP/SC.LT.RMAX) GO TO 55
   45 L = ALOG(SC)/ALOG(RADIX)+.5
      IF (L .EQ. 0) GO TO 55
      FACTOR = DBLE(RADIX)**L
      DO 50 I=1,NN
   50 P(I) = FACTOR*P(I)
C                                           COMPUTE LOWER BOUND ON MODULI OF
C                                             ZEROS.
   55 DO 60 I=1,NN
   60 PT(I) = ABS(SNGL(P(I)))
      PT(NN) = -PT(NN)
C                                           COMPUTE UPPER ESTIMATE OF BOUND
      X = EXP((ALOG(-PT(NN))-ALOG(PT(1)))/N)
      IF (PT(N).EQ.0.) GO TO 65
C                                           IF NEWTON STEP AT THE ORIGIN IS
C                                             BETTER, USE IT.
      XM = -PT(NN)/PT(N)
      IF (XM.LT.X) X = XM
C                                           CHOP THE INTERVAL (0,X) UNTIL FF.LE.0
   65 XM = X*.1
      FF = PT(1)
      DO 70 I=2,NN
   70 FF = FF*XM+PT(I)
      IF (FF.LE.0.) GO TO 75
      X = XM
      GO TO 65
   75 DX = X
C                                           DO NEWTON ITERATION UNTIL X
C                                             CONVERGES TO TWO DECIMAL PLACES
   80 IF (ABS(DX/X).LE..005) GO TO 90
      FF = PT(1)
      DF = FF
      DO 85 I=2,N
         FF = FF*X+PT(I)
         DF = DF*X+FF
   85 CONTINUE
      FF = FF*X+PT(NN)
      DX = FF/DF
      X = X-DX
      GO TO 80
   90 BND = X
C                                           COMPUTE THE DERIVATIVE AS THE INTIAL
C                                             K POLYNOMIAL AND DO 5 STEPS WITH
C                                             NO SHIFT
      NM1 = N-1
      FN = ONE/N
      DO 95 I=2,N
   95 RK(I) = (NN-I)*P(I)*FN
      RK(1) = P(1)
      AA = P(NN)
      BB = P(N)
      ZEROK = RK(N).EQ.ZERO
      DO 115 JJ=1,5
         CC = RK(N)
         IF (ZEROK) GO TO 105
C                                           USE SCALED FORM OF RECURRENCE IF
C                                             VALUE OF K AT 0 IS NONZERO
         T = -AA/CC
         DO 100 I=1,NM1
            J = NN-I
            RK(J) = T*RK(J-1)+P(J)
  100    CONTINUE
         RK(1) = P(1)
         ZEROK = DABS(RK(N)).LE.DABS(BB)*ETA*10.
         GO TO 115
```

```
C                                          USE UNSCALED FORM OF RECURRENCE
  105     DO 110 I=1,NM1
             J = NN-I
             RK(J) = RK(J-1)
  110     CONTINUE
          RK(1) = ZERO
          ZEROK = RK(N).EQ.ZERO
  115 CONTINUE
C                                          SAVE K FOR RESTARTS WITH NEW SHIFTS
      DO 120 I=1,N
  120 TEMP(I) = RK(I)
C                                          LOOP TO SELECT THE QUADRATIC
C                                             CORRESPONDING TO EACH NEW SHIFT
      DO 140 ICNT=1,20
C                                          QUADRATIC CORRESPONDS TO A DOUBLE
C                                             SHIFT TO A NON-REAL POINT AND ITS
C                                             COMPLEX CONJUGATE. THE POINT HAS
C                                             MODULUS BND AND AMPLITUDE ROTATED
C                                             BY 94 DEGREES FROM THE PREVIOUS
C                                             SHIFT
          XXX = COSR*XX-SINR*YY
          YY = SINR*XX+COSR*YY
          XX = XXX
          SR = BND*XX
          SI = BND*YY
          U = -SR-SR
          V = BND*BND
C                                          SECOND STAGE CALCULATION, FIXED
C                                             QUADRATIC
          CALL ZRPQLB (20*ICNT,NZ)
          IF (NZ.EQ.0) GO TO 130
C                                          THE SECOND STAGE JUMPS DIRECTLY TO
C                                             ONE OF THE THIRD STAGE ITERATIONS
C                                             AND RETURNS HERE IF SUCCESSFUL.
C                                          DEFLATE THE POLYNOMIAL, STORE THE
C                                             ZERO OR ZEROS AND RETURN TO THE
C                                             MAIN ALGORITHM.
          J = NDEG-N+1
          JJ = J+NDEG
          Z(J) = SZR
          Z(JJ) = SZI
          NN = NN-NZ
          N = NN-1
          DO 125 I=1,NN
  125     P(I) = QP(I)
          IF (NZ.EQ.1) GO TO 20
          Z(J+1) = RLZR
          Z(JJ+1) = RLZI
          GO TO 20
C                                          IF THE ITERATION IS UNSUCCESSFUL
C                                             ANOTHER QUADRATIC IS CHOSEN AFTER
C                                             RESTORING K
  130     DO 135 I=1,N
  135     RK(I) = TEMP(I)
  140 CONTINUE
C                                          RETURN WITH FAILURE IF NO
C                                             CONVERGENCE WITH 20 SHIFTS
      IER = 131
C                                          CONVERT ZEROS (Z) IN COMPLEX FORM
  145 DO 150 I=1,NDEG
          NPI= NDEG+I
          P(I) = Z(NPI)
  150 CONTINUE
      N2 = NDEG+NDEG
      J = NDEG
      DO 155 I=1,NDEG
          Z(N2-1) = Z(J)
          Z(N2) = P(J)
          N2 = N2-2
          J = J-1
  155 CONTINUE
      IF (IER .EQ. 0) GO TO 9005
C                                          SET UNFOUND ROOTS TO MACHINE INFINITY
      N2 = 2*(NDEG-NN)+3
      DO 160 I=1,N
          Z(N2) = RINFP
          Z(N2+1) = RINFP
          N2 = N2+2
  160 CONTINUE
      GO TO 9000
  165 IER = 129
 9000 CONTINUE
      CALL UERTST (IER,6HZRPOLY)
 9005 RETURN
      END
```

# REQUEST FOR THE SOURCE OF AN IMSL ROUTINE

## RESTRICTIONS ON THE USE OF IMSL

The following points were adapted from the library subscription policies published in the IMSL Library General Information manual, Volume 1. The IMSL routines are proprietary, and the distributor has placed the following restrictions on their usage:

* Only internal usage at one computational site is allowed. IMSL products may not intentionally be made available to other computational sites.

  Arrangements with IMSL must be made if application programs containing embedded object code from IMSL library routines are to be distributed outside Virginia Tech. IMSL, Inc., has the intent of being agreeable in this case if the application programs are to be made available at no charge, but the distributor desires knowledge of such distributions. If the programs are to be made available for a charge, IMSL's contract form (F-A005) covers the required financial arrangement.

* When IMSL library subroutines are used in research work and resulting publications require, in the opinion of the author, the reproduction of listings of pertinent library source code in the research publication, IMSL requests reference, and grants permission. If only the subroutine documentation is necessary or sufficient, IMSL encourages that mode of use. When application programs developed as a by-product of research work contain embedded library routines, these application programs may be taken, as is, by the authors on their departure from the research institution for usage in their new locations. IMSL's only restriction is that any embedded library routines be used only as originally incorporated in the application program and that they not be removed for usage in any other software development.

* IMSL software may not be transferred to other computers without IMSL's specific (by code name) permission.

In many cases graduate students use library elements to aid in satisfying research requirements for their degrees. If, in the opinion of the author, listings of IMSL library source code must be reproduced in the resultant thesis or dissertation in order that the results be duplicatable by others, IMSL grants permission for such reproduction. In cases of such usage, IMSL requests that the author make the following points in the research document in a manner noticeable to someone interested in the listings:

* The listed program is part of a proprietary product belonging to International Mathematical and Statistical Libraries, Inc., in Houston, Texas.

* The listings are reproduced with the permission of IMSL.

* The listings may not be extracted for other purposes or used as the basis for any software development.

I have read the above restrictions and agree to abide by them. Please

furnish a copy the IMSL routine ZRPOLY. signed Mohammad Malakooti-V.

Name Mohammad Malakooti Department Electrical Engineering Phone 552-5549 Hom 961-5156 offi

Computer Center account # to be billed for cost of run 10GE7.

The charge for this service is the cost of the run plus $2.00.

$\lambda_2=1$: E3= 0.190[illegible]9 E2= [illegible] E1= [illegible] E0= 0.15[illegible]
A4= 1.000000 A3=[illegible] A2=-1.0[illegible] A1=-0.291446 A0= 0.816751
ET1=-0.158567 ET2=-0.071468 NU1=-0.475215 NU2= 0.555756 GM1= 1.687101 GM2=-0.740818
PLANT POLES AND ZEROS

| POLES | | ZEROS | |
|---|---|---|---|
| REAL | IMAG | REAL | IMAG |
| 0.922634 | 0.230682 | -0.144123 | 0.942301 |
| 0.922634 | -0.230682 | -0.144123 | -0.942301 |
| -0.761785 | 0.569665 | -0.959909 | 0.0 |
| -0.761785 | -0.569665 | | |

B5= 0.951647 B4= 1.187803 B3= 1.126077 B2= 0.830095 B1= 0.0 B0= 0.0
A6= 1.000000 A5=-0.163530 A4=-0.891490 A3=-0.400287 A2= 0.676672 A1= 0.062905 A0=-0.033752
ET1=-0.158567 ET2=-0.071468 NU1=-0.475215 NU2= 0.555756 GM1= 1.687101 GM2=-0.740818
OVERALL TRANSFER FUNCTION POLES AND ZEROS

| POLES | | ZEROS | |
|---|---|---|---|
| REAL | IMAG | REAL | IMAG |
| 0.197403 | 0.0 | 0.0 | 0.0 |
| -0.259023 | 0.0 | 0.0 | 0.0 |
| -0.729755 | 0.600772 | -0.144123 | 0.942301 |
| -0.729755 | -0.600772 | -0.144123 | -0.942301 |
| 0.842330 | 0.171181 | -0.959909 | 0.0 |
| 0.842330 | -0.171181 | | |

$\lambda_2=10$: E3= 0.820470 E2=-0.321383 E1=-0.319153 E0= 0.712619
A4= 1.000000 A3=-0.322098 A2=-1.002332 A1=-0.291446 A0= 0.816751
ET1=-0.158567 ET2=-0.071468 NU1=-0.475215 NU2= 0.555756 GM1= 1.687101 GM2=-0.740818
PLANT POLES AND ZEROS

| POLES | | ZEROS | |
|---|---|---|---|
| REAL | IMAG | REAL | IMAG |
| 0.922634 | 0.230682 | 0.675252 | 0.670751 |
| 0.922634 | -0.230682 | 0.675252 | -0.670751 |
| -0.761785 | 0.569665 | -0.958799 | 0.0 |
| -0.761785 | -0.569665 | | |

B5= 4.102349 B4=-1.606913 B3=-1.595765 B2= 3.563096 B1= 0.0 B0= 0.0
A6= 1.000000 A5=-0.163530 A4=-0.592039 A3=-1.082109 A2= 0.727825 A1= 0.625014 A0=-0.337529
ET1=-0.158567 ET2=-0.071468 NU1=-0.475215 NU2= 0.555756 GM1= 1.687101 GM2=-0.740818
OVERALL TRANSFER FUNCTION POLES AND ZERCS

| POLES | | ZEROS | |
|---|---|---|---|
| REAL | IMAG | REAL | IMAG |
| 0.513095 | 0.0 | 0.0 | 0.0 |
| -0.717872 | 0.0 | 0.0 | 0.0 |
| -0.645517 | 0.926340 | 0.675252 | 0.670752 |
| -0.645517 | -0.926340 | 0.675252 | -0.670752 |
| 0.829670 | 0.175589 | -0.958798 | 0.0 |
| 0.829670 | -0.175589 | | |

ORIGINAL PAGE IS
OF POOR QUALITY

```
C      UNIT CIRCLE:
C      THIS PROGRAM  DRAWS A UNIT CIRCLE AND LOCATES POLES & ZEROS OF A
C      SYSTEM IN IT.
       DIMENSION X1(8001),X2(8001),X3(8001),X4(8001),Y1(8001),Y2(8001),Y3
      +(8001),Y4(8001),XP(4),YP(4),XZ(3),YZ(3),AXZ(3),AYZ(3)
       DATA X1(1),X2(1),X3(1),X4(1)/0.0,4.0,0.0,-4.0/
       DATA Y1(1),Y2(1),Y3(1),Y4(1)/4.0,0.0,-4.0,0.0/
       DATA XP/0.922834,0.922834,-0.761785,-0.761785/
       DATA YP/0.230662,-0.230662,0.569665,-0.569665/
       DATA XZ/-0.144123,-0.144123,-0.959908/
       DATA YZ/0.942301,-0.942301,0.0/
       N=8000
       YMAX=1.0
       YMIN=-1.0
       DO 25 I=2,8001
       X2(I)=X2(I-1)-0.0005
       Y2(I)=-ABS(SQRT(16.0-X2(I)**2))
       J=8003-I
       X1(J)=X2(I-1)
       Y1(J)=-Y2(I-1)
       X3(J)=-X2(I-1)
       Y3(J)=Y2(I-1)
       X4(I)=-X2(I)
       Y4(I)=-Y2(I)
  25   CONTINUE
       CALL PLOTS(0.0,0.0,8001)
       CALL PLOT(5.0,6.0,-3)
       CALL FACTOR(0.65)
       XSTART=-1.0
       DT=1/4.0
       CALL AXIS (-4.0,0.0,'REAL',-4,8.0,0.0,XSTART,DT)
       YT=ABS(YMAX)
       IF(ABS(YMIN).GT.YT) YT=ABS(YMIN)
       YSTART=-YT
       DY=YT/4.0
       CALL AXIS (0.0,-4.0,'IMAGINARY',9,8.0,90.0,YSTART,DY)
       CALL NEWPEN(3)
       CALL PLOT(X1(1),Y1(1),3)
       DO 20 I=2,N
  20   CALL PLOT(X1(I),Y1(I),2)
       CALL PLOT(X2(1),Y2(1),3)
       DO 30 I=2,N
  30   CALL PLOT(X2(I),Y2(I),2)
       CALL PLOT(X3(1),Y3(1),3)
       DO 40 I=2,N
  40   CALL PLOT(X3(I),Y3(I),2)
       CALL PLOT(X4(1),Y4(1),3)
       DO 50 I=2,N
  50   CALL PLOT(X4(I),Y4(I),2)
       DX=0.25
       DY=0.25
       DO 100 I=1,4
       CALL SYMBOL(XP(I)/DX,YP(I)/DY,0.2,4,0.0,-1)
 100   CONTINUE
       DO 200 J=1,3
       AXZ(J)=XZ(J)/DX-0.01515*4.0
       AYZ(J)=YZ(J)/DY-0.02727*4.0
       CALL SYMBOL(AXZ(J),AYZ(J),0.2,112,0.0,-1)
 200   CONTINUE
       CALL PLOT(18.0,0.0,-3)
       CALL PLOT(0.0,0.0,+999)
       STOP
       END
```

```
C     UNIT CIRCLE:
C     THIS PROGRAM  DRAWS A UNIT CIRCLE AND LOCATES POLES & ZEROS OF A
C     SYSTEM IN IT.
      DIMENSION X1(8001),X2(8001),X3(8001),X4(8001),Y1(8001),Y2(8001),Y3
     +(8001),Y4(8001),XP(6),YP(6),XZ(5),YZ(5),AXZ(5),AYZ(5)
      DATA X1(1),X2(1),X3(1),X4(1)/0.0,4.0,0.0,-4.0/
      DATA Y1(1),Y2(1),Y3(1),Y4(1)/4.0,0.0,-4.0,0.0/
      DATA XP/0.197403,-0.259023,-0.729755,-0.729755,0.842350,0.842350/
      DATA YP/0.0,0.0,0.600772,-0.600772,0.171181,-0.171181/
      DATA XZ/0.0,0.0,-0.144123,-0.144123,-0.959909/
      DATA YZ/0.0,0.0,0.942301,-0.942301,0.0/
      N=8000
      YMAX=1.0
      YMIN=-1.0
      DO 25 I=2,8001
      X2(I)=X2(I-1)-0.0005
      Y2(I)=-ABS(SQRT(16.0-X2(I)**2))
      J=8003-I
      X1(J)=X2(I-1)
      Y1(J)=-Y2(I-1)
      X3(J)=-X2(I-1)
      Y3(J)=Y2(I-1)
      X4(I)=-X2(I)
      Y4(I)=-Y2(I)
   25 CONTINUE
      CALL PLOTS(0.0,0.0,8001)
      CALL PLOT(5.0,6.0,-3)
      CALL FACTOR(0.65)
      XSTART=-1.0
      DT=1/4.0
      CALL AXIS (-4.0,0.0,'REAL',-4,8.0,0.0,XSTART,DT)
      YT=ABS(YMAX)
      IF(ABS(YMIN).GT.YT) YT=ABS(YMIN)
      YSTART=-YT
      DY=YT/4.0
      CALL AXIS (0.0,-4.0,'IMAGINARY',9,8.0,90.0,YSTART,DY)
      CALL NEWPEN(3)
      CALL PLOT(X1(1),Y1(1),3)
      DO 20 I=2,N
   20 CALL PLOT(X1(I),Y1(I),2)
      CALL PLOT(X2(1),Y2(1),3)
      DO 30 I=2,N
   30 CALL PLOT(X2(I),Y2(I),2)
      CALL PLOT(X3(1),Y3(1),3)
      DO 40 I=2,N
   40 CALL PLOT(X3(I),Y3(I),2)
      CALL PLOT(X4(1),Y4(1),3)
      DO 50 I=2,N
   50 CALL PLOT(X4(I),Y4(I),2)
      DX=0.25
      DY=0.25
      DO 100 I=1,6
      CALL SYMBOL(XP(I)/DX,YP(I)/DY,0.2,4,0.0,-1)
  100 CONTINUE
      DO 200 J=1,5
      AXZ(J)=XZ(J)/DX-0.01515*4.0
      AYZ(J)=YZ(J)/DY-0.02727*4.0
      CALL SYMBOL(AXZ(J),AYZ(J),0.2,112,0.0,-1)
  200 CONTINUE
      CALL PLOT(16.0,0.0,-3)
      CALL PLOT(0.0,0.0,+999)
      STOP
      END
```

## Appendix D: Overall Transfer Functions for Example 6

Contents:



Note: a's and b's in printout correspond to p's and q's in (3-48) and (3-61)

**Consider the control configuration in Fig. 3-12 a (or c) with a convergent identifier, i.e. the time-invariant, asymptotic case. For the plant in (3-17) with the controller in (3-46) and the actuator of (3-45) with**

$$g \triangleq e^{-\sigma T} \tag{D-1}$$

then, since

$$U_c(z) = \frac{\delta\ R(z) + (\nu_1\ z^{-1} + \nu_2\ z^{-2})Y(z)}{(1 - \eta_1\ z^{-1} - \eta_2\ z^{-2})} \tag{D-2}$$

and

$$\frac{Y(z)}{U_c(z)} = \frac{U(z)}{U_c(z)} \qquad \frac{Y(z)}{U(z)}, \tag{D-3}$$

$$\frac{Y(z)}{U_c(z)} = \frac{(1 - \eta_1 z^{-1} - \eta_2 z^{-2})\ Y(z)}{\delta\ R(z) + (\nu_1 z^{-1} + \nu_2 z^{-2})\ Y(z)}$$

$$= \left[\frac{\beta_{11} z + \beta_{12}}{z^2 - \alpha_{11} z - \alpha_{12}} + \frac{\beta_{21} z + \beta_{22}}{z^2 - \alpha_{21} z - \alpha_{22}}\right] \left(\frac{1 - g}{z - g}\right). \tag{D-4}$$

Cross-multiplying in (D-4) and rearranging yields

$$\frac{Y(z)}{R(z)} = \frac{z^2 \left\{\left[(\beta_{11} z + \beta_{12})(z^2 - \alpha_{21} z - \alpha_{22}) + (\beta_{11} z + \beta_{12})(z^2 - \alpha_{11} z - \alpha_{12})\right]\delta(1 - g)\right\}}{\left\{(z - g)(z^2 - \alpha_{11} z - \alpha_{12})(z^2 - \alpha_{21} z - \alpha_{22})(z^2 - \eta_1 z - \eta_2) - (\nu_1 + \nu_2)(1 - g)\ \left[(\beta_{11} z + \beta_{12})(z^2 - \alpha_{21} z - \alpha_{22}) + (\beta_{11} z + \beta_{12})(z^2 - \alpha_{11} z - \alpha_{12})\right]\right\}}$$

$$= \frac{(1 - g)\ N\ (z)}{(z - g)\ D_1(z) - (1 - g)\ D_2(z)} = \frac{q_5\ z^5 + q_4 z^4 + q_3 z^3 + q_2 z^2 + q_1 z + q_0}{[p_7 z^7 + p_6 z^6 + p_5 z^5 + p_4 z^4 + p_3 z^3 + p_2 z^2 + p_1 z + p_0]}, \tag{D-5}$$

where $N(z)$, $D_1(z)$, and $D_2(z)$ are from (B-2), (B-4), and (B-5), respectively.

Note that the $q_i$ of (D-5) are equal to the $(1-g)\ b_i$ of (3-32) - (3-36) as shown in (3-49) - (3-53). From (B-6) and (B-7) the $p_i$ of (3-54) - (3-60) are readily formed.

Similarly for the control configuration of Fig. 3-12b after identifier convergence, from moving the pickoff point across the actuator

$$U_c(z) = \delta\ R(z) + [\eta_1 (\frac{1-g}{z-g})\ z^{-1} + \eta_2 (\frac{1-g}{z-g})\ z^{-2}]\ U_c(z) \\ + [\nu_1\ z^{-1} + \nu_2\ z^{-2}]\ Y(z) \qquad \text{(D-6)}$$

and

$$\frac{Y(z)}{U_c(z)} = (\frac{1-g}{z-g})\ \frac{Y(z)}{U(z)}\ .$$

Therefore

$$\frac{[z^3 - gz^2 - \eta_1\ (1-g)\ z - \eta_2\ (1-g)]\ Y(z)}{z^2 \delta\ R(z) + [\nu_1 z + \nu_2]\ Y(z)} = (1-g) \left[\frac{\beta_{11} z + \beta_{12}}{z^2 - \alpha_{11} z - \alpha_{12}} + \frac{\beta_{21} z + \beta_{22}}{z^2 - \alpha_{21} z - \alpha_{22}}\right]. \qquad \text{(D-7)}$$

Again cross-multiplying and rearranging yields

$$\frac{Y(z)}{R(z)} = \frac{\left\{ z^2\ \delta\ (1-g)[(\beta_{11} z + \beta_{12})(z^2 - \alpha_{21} z - \alpha_{22}) + (\beta_{21} z + \beta_{22})(z^2 - \alpha_{11} z - \alpha_{12})] \right\}}{\left\{ [z^3 - gz^2 - \eta_1\ (1-g) z - \eta_2 (1-g)](z^2 - \alpha_{11} z - \alpha_{12})(z^2 - \alpha_{21} z - \alpha_{22}) - [\nu_1 z + \nu_2](1-g)[(\beta_{11} z + \beta_{12})(z^2 - \alpha_{21} z - \alpha_{22}) + (\beta_{21} z + \beta_{22})(z^2 - \alpha_{11} z - \alpha_{12})] \right\}}$$

$$= \frac{(1-g)\ N(z)}{\left\{ [z^3 - gz^2 - \eta_1 (1-g)\ z - \eta_2 (1-g)](z^2 - \alpha_{11} z - \alpha_{12})(z^2 - \alpha_{21} z - \alpha_{22}) - (1-g) D_2(z) \right\}} \qquad \text{(D-8)}$$

where $N(z)$ and $D_2(z)$ are from (B-2) and (B-7), respectively. Expansion yields (3-62) - (3-68).

```
      DIMENSION G1(2),F1(2),ZET(2),W(2),AL(2),G2(2),F2(2)
      DATA T/0.5/,AL/1.0,1.0/,ZET/0.2,0.02/,W/0.5,5.0/,D/0.5/,ZETA/0.6/
      DATA Y1,Y2,Y3,Y4,Y5,U1,U2,U3,U4,U5/0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
     +,0.0,0.0/,SIG/1.0/
      DATA YC1,YC2,YC3,YC4,YC5,YC6,YC7/0.0,0.0,0.0,0.0,0.0,0.0,0.0/,R1,R
     +2,R3,R4,R5,R6,R7,R/0.0,0.0,0.0,0.0,0.0,0.0,0.0,1.0/
      DO 10 I=1,2
      G1(I)=2*EXP(-ZET(I)*W(I)*T)*COS(W(I)*T*SQRT(1-ZET(I)**2))
      IF(I.EQ.1)A11=G1(1)
      IF(I.EQ.2)A21=G1(2)
      G2(I)=-EXP(-2*ZET(I)*W(I)*T)
      IF(I.EQ.1)A12=G2(1)
      IF(I.EQ.2)A22=G2(2)
      F1(I)=(AL(I)/W(I)**2)*(1-EXP(-ZET(I)*W(I)*T)*(COS(W(I)*T*SQRT(1-ZE
     +T(I)**2))+(ZET(I)/SQRT(1-ZET(I)**2))*SIN(W(I)*T*SQRT(1-ZET(I)**2))
     +))
      IF(I.EQ.1)B11=F1(1)
      IF(I.EQ.2)B21=F1(2)
      F2(I)=(AL(I)/W(I)**2)*(EXP(-ZET(I)*W(I)*T)*(EXP(-ZET(I)*W(I)*T)-CO
     +S(W(I)*T*SQRT(1-ZET(I)**2))+(ZET(I)/SQRT(1-ZET(I)**2))*SIN(W(I)*T*
     +SQRT(1-ZET(I)**2))))
      IF(I.EQ.1)B12=F2(1)
      IF(I.EQ.2)B22=F2(2)
10    CONTINUE
      GM1=2*EXP(-ZETA*W(1)*T)*COS(W(1)*T*SQRT(1.0-ZETA**2))
      GM2=-EXP(-2*ZETA*W(1)*T)
      ET1=GM1-A11
      ANU1=(ET1*A12+(ET1*A11-A12+GM2)*(A11-A12*B11/B12))/(B11*A11+B12-A1
     +2*B11**2/B12)
      ET2=A11*ET1+GM2-B11*ANU1-A12
      ANU2=ET2*A12/B12
      G=EXP(-SIG*T)
      B5=D*(1.0-G)*(B11+B21)
      B4=D*(1.0-G)*(-B11*A21+B12-B21*A11+B22)
      B3=D*(1.0-G)*(-B11*A22-B12*A21-B21*A12-B22*A11)
      B2=D*(1.0-G)*(-B12*A22-B22*A12)
      B1=0.0
      B0=B1
      A7=1.0
      A6=-G-A21-A11-ET1
      A5=(G*(A21+A11+ET1)-A22+A11*A21-A12+ET1*A21+ET1*A11-ET2)
      A4=(-G*(-A22+A11*A21-A12+ET1*A21+ET1*A11-ET2)+A11*A22+A12*A21+ET1
     +*A22-ET1*A11*A21+ET1*A12+ET2*A21+ET2*A11-ANU1*(B11+B21)*(1.0-G))
      A3=(-G*(A11*A22+A12*A21+ET1*A22-ET1*A11*A21+ET1*A12+ET2*A21+ET2*A1
     +1)+A12*A22-ET1*A11*A22-ET1*A12*A21+ET2*A22-ET2*A11*A21+ET2*A12-ANU
     +1*(-B11*A21+B22+B12-B21*A11)*(1.0-G)-ANU2*(B11+B21)*(1.0-G))
      A2=(-G*(A12*A22-ET1*A11*A22-ET1*A12*A21+ET2*A22-ET2*A11*A21+ET2*A1
     +2)-ET1*A12*A22-ET2*A11*A22-ET2*A12*A21+ANU1*(B11*A22+B12*A21+B21*A
     +12+B22*A11)*(1.0-G)-ANU2*(-B11*A21+B22+B12-B21*A11)*(1.0-G))
      A1=(G*(ET1*A12*A22+ET2*A11*A22+ET2*A12*A21)-ET2*A12*A22+ANU2*(B11*
     +A22+B12*A21+B21*A12+B22*A11)*(1.0-G)+ANU1*(B12*A22+B22*A12)*(1.0-
     +G))
      A0=G*ET2*A12*A22+ANU2*(B12*A22+B22*A12)*(1.0-G)
      WRITE(6,60)B5,B4,B3,B2,B1,B0
60    FORMAT(1X,'B5=',F9.6,2X,'B4=',F9.6,2X,'B3=',F9.6,2X,'B2=',F9.6,2X,
     +'B1=',F9.6,2X,'B0=',F9.6,2X)
      WRITE(6,70)A7,A6,A5,A4,A3,A2,A1,A0
70    FORMAT(1X,'A7=',F9.6,2X,'A6=',F9.6,2X,'A5=',F9.6,2X,'A4=',F9.6,2X,
     +'A3=',F9.6,2X,'A2=',F9.6,2X,'A1=',F9.6,2X,'A0=',F9.6,2X)
      WRITE(6,30)ET1,ET2,ANU1,ANU2,GM1,GM2
30    FORMAT(1X,'ET1=',F9.6,2X,'ET2=',F9.6,2X,'NU1=',F9.6,2X,'NU2=',F9.6
     +,2X,'GM1=',F9.6,2X,'GM2=',F9.6,2X)
      WRITE(6,100)
100   FORMAT(1X,'Y1:CONTROLLED SYS.',3X,'Y2:OVERALL T.F.',5X,'(Y1-Y2)**2
     +')
      DO 55 J=1,100
      Y=(G+A21+A11)*Y1-(G*(A21+A11)-A22+A11*A21-A12)*Y2-(A11*A22+A12*A21
     +-G*(-A22+A11*A21-A12))*Y3-(A12*A22-G*(A11*A22+A12*A21))*Y4+G*A12*
     +A22*Y5+(1.0-G)*(B11+B21)*U2+(1.0-G)*(-B11*A21+B22+B12-B21*A11)*U3+
     +(1.0-G)*(-B11*A22-B12*A21-B21*A12-B22*A11)*U4+(1.0-G)*(-B12*A22-B2
     +2*A12)*U5
      U=D*R+ET1*U1+ET2*U2+ANU1*Y1+ANU2*Y2
      Y5=Y4
      Y4=Y3
      Y3=Y2
      Y2=Y1
      Y1=Y
      U5=U4
      U4=U3
      U3=U2
      U2=U1
      U1=U
```

```
      YC=-A6*YC1-A5*YC2-A4*YC3-A3*YC4-A2*YC5-A1*YC6-A0*YC7+B5*R2+B4*R3+B
     *3*R4+B2*R5+B1*R6+B0*R7
      YC7=YC6
      YC6=YC5
      YC5=YC4
      YC4=YC3
      YC3=YC2
      YC2=YC1
      YC1=YC
      R7=R6
      R6=R5
      R5=R4
      R4=R3
      R3=R2
      R2=R1
      R1=R
      DY=(Y-YC)**2
      WRITE(6,65)Y,YC,DY
65    FORMAT(1X,F13.8,9X,F13.8,5X,F13.8,5X)
35    CONTINUE
      STOP
      END
```

```
B5= 0.037444   B4= 0.040730   B3= 0.044380   B2= 0.032062
B1= 0.000000   B0= 0.000000
A7= 1.000000   A6=-0.770061   A5=-0.862752   A4= 0.157761
A3= 0.990814   A2=-0.325874   A1=-0.025889   A0=-0.071794
GM1= 1.067101   GM2=-0.740518
ET1=-0.156567   ET2=-0.071466   NU1=-0.472215   NU2= 0.555756
```

| Y1:CONTROLLED SYS. | Y2:OVERALL T.F. | (Y1-Y2)**2 |
|---|---|---|
| 0.00000000 | 0.00000000 | 0.00000000 |
| 0.00000000 | 0.00000000 | 0.00000000 |
| 0.03744444 | 0.03744444 | 0.00000000 |
| 0.11301530 | 0.11301530 | 0.00000000 |
| 0.24665000 | 0.24665000 | 0.00000000 |
| 0.44656180 | 0.44656160 | 0.00000000 |
| 0.66967510 | 0.66967480 | 0.00000000 |
| 0.93212080 | 0.93212010 | 0.00000000 |
| 1.19115400 | 1.19115300 | 0.00000000 |
| 1.43985000 | 1.43984900 | 0.00000000 |
| 1.67098900 | 1.67098600 | 0.00000000 |
| 1.85516400 | 1.85518200 | 0.00000000 |
| 2.01068400 | 2.01068200 | 0.00000000 |
| 2.11737600 | 2.11737500 | 0.00000000 |
| 2.18532000 | 2.18532100 | 0.00000000 |
| 2.22514800 | 2.22515000 | 0.00000000 |
| 2.22875500 | 2.22875900 | 0.00000000 |
| 2.22229700 | 2.22230100 | 0.00000000 |
| 2.19892300 | 2.19892700 | 0.00000000 |
| 2.17125900 | 2.17126400 | 0.00000000 |
| 2.14916500 | 2.14916300 | 0.00000000 |
| 2.12490200 | 2.12490600 | 0.00000000 |
| 2.11508500 | 2.11508700 | 0.00000000 |
| 2.10947700 | 2.10947900 | 0.00000000 |
| 2.11212800 | 2.11213200 | 0.00000000 |
| 2.12521500 | 2.12521700 | 0.00000000 |
| 2.13777700 | 2.13778300 | 0.00000000 |
| 2.15872900 | 2.15873500 | 0.00000000 |
| 2.17743700 | 2.17744600 | 0.00000000 |
| 2.19540500 | 2.19541600 | 0.00000000 |
| 2.21393500 | 2.21394600 | 0.00000000 |
| 2.22497800 | 2.22499100 | 0.00000000 |
| 2.23655800 | 2.23657200 | 0.00000000 |
| 2.24204000 | 2.24205400 | 0.00000000 |
| 2.24422500 | 2.24424000 | 0.00000000 |
| 2.24562000 | 2.24564000 | 0.00000000 |
| 2.24201600 | 2.24203100 | 0.00000000 |
| 2.24011100 | 2.24012200 | 0.00000000 |
| ⋮ | ⋮ | ⋮ |
| 2.23283500 | 2.23284400 | 0.00000000 |
| 2.23288700 | 2.23289600 | 0.00000000 |
| 2.23266700 | 2.23267500 | 0.00000000 |
| 2.23283200 | 2.23284200 | 0.00000000 |

```
      DIMENSION G1(2),F1(2),ZET(2),W(2),AL(2),G2(2),F2(2)
      DATA T/0.5/,AL/1.0,1.0/,ZET/0.2,0.02/,W/0.5,5.0/,D/0.5/,ZETA/0.6/
      DATA Y1,Y2,Y3,Y4,Y5,U1,U2,U3,U4,U5/0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
     +,0.0,0.0/,SIG/1.0/
      DATA YC1,YC2,YC3,YC4,YC5,YC6,YC7/0.0,0.0,0.0,0.0,0.0,0.0,0.0/,R1,R
     +2,R3,R4,R5,R6,R7,R/0.0,0.0,0.0,0.0,0.0,0.0,0.0,1.0/
      DO 10 I=1,2
      G1(I)=2*EXP(-ZET(I)*W(I)*T)*COS(W(I)*T*SQRT(1-ZET(I)**2))
      IF(I.EQ.1)A11=G1(1)
      IF(I.EQ.2)A21=G1(2)
      G2(I)=-EXP(-2*ZET(I)*W(I)*T)
      IF(I.EQ.1)A12=G2(1)
      IF(I.EQ.2)A22=G2(2)
      F1(I)=(AL(I)/W(I)**2)*(1-EXP(-ZET(I)*W(I)*T)*(COS(W(I)*T*SQRT(1-ZE
     +T(I)**2))+(ZET(I)/SQRT(1-ZET(I)**2))*SIN(W(I)*T*SQRT(1-ZET(I)**2))
     +))
      IF(I.EQ.1)B11=F1(1)
      IF(I.EQ.2)B21=F1(2)
      F2(I)=(AL(I)/W(I)**2)*(EXP(-ZET(I)*W(I)*T)*(EXP(-ZET(I)*W(I)*T)-CO
     +S(W(I)*T*SQRT(1-ZET(I)**2))+(ZET(I)/SQRT(1-ZET(I)**2))*SIN(W(I)*T*
     +SQRT(1-ZET(I)**2))))
      IF(I.EQ.1)B12=F2(1)
      IF(I.EQ.2)B22=F2(2)
   10 CONTINUE
      GM1=2*EXP(-ZETA*W(1)*T)*COS(W(1)*T*SQRT(1.0-ZETA**2))
      GM2=-EXP(-2*ZETA*W(1)*T)
      ET1=GM1-A11
      ANU1=(ET1*A12+(ET1*A11-A12+GM2)*(A11-A12*B11/B12))/(B11*A11+B12-A1
     +2*B11**2/B12)
      ET2=A11*ET1+GM2-B11*ANU1-A12
      ANU2=ET2*A12/B12
      G=EXP(-SIG*T)
      B5=D*(1.0-G)*(B11+B21)
      B4=D*(1.0-G)*(-B11*A21+B12-B21*A11+B22)
      B3=D*(1.0-G)*(-B11*A22-B12*A21-B21*A12-B22*A11)
      B2=D*(1.0-G)*(-B12*A22-B22*A12)
      B1=0.0
      B0=B1
      A7=1.0
      A6=-A21-A11-G
      A5=-A22+A11*A21-A12+G*(A21+A11)-ET1*(1.0-G)
      A4=A11*A22+A12*A21-G*(-A22+A11*A21-A12)+ET1*(1.0-G)*(A21+A11)-ET2*
     +(1.0-G)-ANU1*(1.0-G)*(B11+B21)
      A3=A12*A22-G*(A11*A22+A12*A21)-ET1*(1.0-G)*(-A22+A11*A21-A12)-ET2*
     +(1.0-G)*(-A21-A11)-ANU1*(1.0-G)*(-B11*A21+B22+B12-B21*A11)-ANU2*(1
     +.0-G)*(B11+B21)
      A2=-G*A12*A22-ET1*(1.0-G)*(A11*A22+A12*A21)-ET2*(1.0-G)*(-A22+A11*
     +A21-A12)+ANU1*(1.0-G)*(B11*A22+B12*A21+B21*A12+B22*A11)-ANU2*(1.0-
     +G)*(-B11*A21+B22+B12-B21*A11)
      A1=-ET1*(1.0-G)*A12*A22-ET2*(1.0-G)*(A11*A22+A12*A21)+ANU2*(1.0-G)
     +*(B11*A22+B12*A21+B21*A12+B22*A11)+ANU1*(1.0-G)*(B12*A22+B22*A12)
      A0=-ET2*(1.0-G)*A12*A22+ANU2*(1.0-G)*(B12*A22+B22*A12)
      WRITE(6,60)B5,B4,B3,B2,B1,B0
   60 FORMAT(1X,'B5=',F9.6,2X,'B4=',F9.6,2X,'B3=',F9.6,2X,'B2=',F9.6,2X,
     +'B1=',F9.6,2X,'B0=',F9.6,2X)
      WRITE(6,70)A7,A6,A5,A4,A3,A2,A1,A0
   70 FORMAT(1X,'A7=',F9.6,2X,'A6=',F9.6,2X,'A5=',F9.6,2X,'A4=',F9.6,2X,
     +'A3=',F9.6,2X,'A2=',F9.6,2X,'A1=',F9.6,2X,'A0=',F9.6,2X)
      WRITE(6,30)ET1,ET2,ANU1,ANU2,GM1,GM2
   30 FORMAT(1X,'ET1=',F9.6,2X,'ET2=',F9.6,2X,'NU1=',F9.6,2X,'NU2=',F9.6
     +,2X,'GM1=',F9.6,2X,'GM2=',F9.6,2X)
      WRITE(6,100)
  100 FORMAT(1X,'Y1:CONTROLLED SYS.',3X,'Y2:OVERALL T.F.',3X,'(Y1-Y2)**2
     +')
      DO 35 J=1,100
      Y=(G+A21+A11)*Y1-(G*(A21+A11)-A22+A11*A21-A12)*Y2-(A11*A22+A12*A21
     +-G*(-A22+A11*A21-A12))*Y3-(A12*A22-G*(A11*A22+A12*A21))*Y4+G*A12*
     +A22*Y5+(1.0-G)*(B11+B21)*U2+(1.0-G)*(-B11*A21+B22+B12-B21*A11)*U3+
     +(1.0-G)*(-B11*A22-B12*A21-B21*A12-B22*A11)*U4+(1.0-G)*(-B12*A22-B2
     +2*A12)*U5
      U=D*R-D*G*R1+ANU1*Y1-ANU1*G*Y2+ANU2*Y2-ANU2*G*Y3+G*U1+ET1*(1.0-G)*
     +U2+ET2*(1.0-G)*U3
      Y5=Y4
      Y4=Y3
      Y3=Y2
      Y2=Y1
      Y1=Y
      U5=U4
      U4=U3
      U3=U2
      U2=U1
      U1=U
```

```
      YC=-A6*YC1-A5*YC2-A4*YC3-A3*YC4-A2*YC5-A1*YC6-A0*YC7+B5*R2+B4*R3+B
     +3*R4+B2*R5+B1*R6+B0*R7
      YC7=YC6
      YC6=YC5
      YC5=YC4
      YC4=YC3
      YC3=YC2
      YC2=YC1
      YC1=YC
      R7=R6
      R6=R5
      R5=R4
      R4=R3
      R3=R2
      R2=R1
      R1=R
      DY=(Y-YC)**2
      WRITE(6,65)Y,YC,DY
65    FORMAT(1X,F13.8,9X,F13.8,5X,F13.8,5X)
35    CONTINUE
      STOP
      END
```

```
 B5= 0.037444   B4= 0.046736   B3= 0.044366   B2= 0.032662

 B1= 0.000000   B0= 0.000000

 A7= 1.000000   A6=-0.928628   A5=-0.744578   A4= 0.360112

 A3= 0.926707   A2=-0.552717   A1= 0.024593   A0=-0.013281

 GM1= 1.687101  GM2=-0.740818

 ET1=-0.158567  ET2=-0.071468  NU1=-0.475215  NU2= 0.555756
```

| Y1:CONTROLLED SYS. | Y2:OVERALL T.F. | (Y1-Y2)**2 |
|---|---|---|
| 0.00000000 | 0.00000000 | 0.00000000 |
| 0.00000000 | 0.00000000 | 0.00000000 |
| 0.03744444 | 0.03744444 | 0.00000000 |
| 0.11895280 | 0.11895280 | 0.00000000 |
| 0.26691040 | 0.26691040 | 0.00000000 |
| 0.48417500 | 0.48417480 | 0.00000000 |
| 0.73204700 | 0.73204660 | 0.00000000 |
| 1.01587800 | 1.01587700 | 0.00000000 |
| 1.29278600 | 1.29278500 | 0.00000000 |
| 1.55093600 | 1.55093200 | 0.00000000 |
| 1.78245800 | 1.78245000 | 0.00000000 |
| 1.96054600 | 1.96053200 | 0.00000000 |
| 2.10239800 | 2.10238100 | 0.00000000 |
| 2.19349000 | 2.19346800 | 0.00000000 |
| 2.24466100 | 2.24463400 | 0.00000000 |
| 2.26917900 | 2.26915200 | 0.00000000 |
| 2.26195600 | 2.26192700 | 0.00000000 |
| 2.24776500 | 2.24773500 | 0.00000000 |
| 2.22218400 | 2.22215100 | 0.00000000 |
| 2.19566700 | 2.19563200 | 0.00000000 |
| 2.17730100 | 2.17726600 | 0.00000000 |
| 2.15895200 | 2.15891300 | 0.00000000 |
| 2.15415200 | 2.15411300 | 0.00000000 |
| 2.15333200 | 2.15329200 | 0.00000000 |
| 2.15841000 | 2.15836800 | 0.00000000 |
| 2.17140600 | 2.17136700 | 0.00000000 |
| ⋮ | ⋮ | ⋮ |
| 2.23267900 | 2.23284000 | 0.00000000 |
| 2.23287500 | 2.23283600 | 0.00000000 |
| 2.23281900 | 2.23277600 | 0.00000000 |
| 2.23291500 | 2.23287200 | 0.00000000 |
| 2.23282200 | 2.23277700 | 0.00000000 |
| 2.23287600 | 2.23283200 | 0.00000000 |
| 2.23287300 | 2.23282900 | 0.00000000 |
| 2.23282500 | 2.23278200 | 0.00000000 |
| 2.23289800 | 2.23285600 | 0.00000000 |
| 2.23282400 | 2.23278600 | 0.00000000 |
| 2.23286700 | 2.23282900 | 0.00000000 |
| 2.23286500 | 2.23283000 | 0.00000000 |
| 2.23282800 | 2.23279300 | 0.00000000 |

**Appendix E:** **Factorization of (3-48) and (3-61) for various $\lambda_2$ as in Table 3-1**

Contents:



Note: a's and b's correspond to p's and q's in (3-48) and (3-61).

```
C     THIS PROGRAM DETERMINES CONTROLLED SYSTEM PARS. AND SINGS.
      DIMENSION G1(2),F1(2),ZET(2),W(2),AL(2),G2(2),F2(2)
      COMPLEX Z(7),Z1(5)
      DATA T/0.5/,AL/1.0,1.0/,ZET/0.2,0.02/,W/0.5,[illegible].0/,D/0.5/,ZETA/0.6/
     +,SIG/1.0/,PI/3.1415926/
      DO 10 I=1,2
      G1(I)=2*EXP(-ZET(I)*W(I)*T)*COS(W(I)*T*SQRT(1-ZET(I)**2))
      IF(I.EQ.1)A11=G1(1)
      IF(I.EQ.2)A21=G1(2)
      G2(I)=-EXP(-2*ZET(I)*W(I)*T)
      IF(I.EQ.1)A12=G2(1)
      IF(I.EQ.2)A22=G2(2)
      F1(I)=(AL(I)/W(I)**2)*(1-EXP(-ZET(I)*W(I)*T)*(COS(W(I)*T*SQRT(1-ZE
     +T(I)**2))+(ZET(I)/SQRT(1-ZET(I)**2))*SIN(W(I)*T*SQRT(1-ZET(I)**2))
     +))
      IF(I.EQ.1)B11=F1(1)
      IF(I.EQ.2)B21=F1(2)
      F2(I)=(AL(I)/W(I)**2)*(EXP(-ZET(I)*W(I)*T)*(EXP(-ZET(I)*W(I)*T)-CO
     +S(W(I)*T*SQRT(1-ZET(I)**2))+(ZET(I))/SQRT(1-ZET(I)**2))*SIN(W(I)*T*
     +SQRT(1-ZET(I)**2))))
      IF(I.EQ.1)B12=F2(1)
      IF(I.EQ.2)B22=F2(2)
   10 CONTINUE
      GM1=2*EXP(-ZETA*W(1)*T)*COS(W(1)*T*SQRT(1.0-ZETA**2))
      GM2=-EXP(-2*ZETA*W(1)*T)
      ET1=GM1-A11
      ANU1=(ET1*A12+(ET1*A11-A12+GM2)*(A11-A12*B11/B12))/(B11*A11+B12-A1
     +2*B11**2/B12)
      ET2=A11*ET1+GM2-B11*ANU1-A12
      ANU2=ET2*A12/B12
      G=EXP(-SIG*T)
      B5=D*(1.0-G)*(B11+B21)
      B4=D*(1.0-G)*(-B11*A21+B12-B21*A11+B22)
      B3=D*(1.0-G)*(-B11*A22-B12*A21-B21*A12-B22*A11)
      B2=D*(1.0-G)*(-B12*A22-B22*A12)
      B1=0.0
      B0=B1
      A7=1.0
      A6=-G-A21-A11-ET1
      A5=(G*(A21+A11+ET1)-A22+A11*A21-A12+ET1*A21+ET1*A11-ET2)
      A4=(-G*(-A22+A11*A21-A12+ET1*A21+ET1*A11-ET2)+A11*A22+A12*A21+ET1
     +*A22-ET1*A11*A21+ET1*A12+ET2*A21+ET2*A11-ANU1*(B11+B21)*(1.0-G))
      A3=(-G*(A11*A22+A12*A21+ET1*A22-ET1*A11*A21+ET1*A12+ET2*A21+ET2*A1
     +1)+A12*A22-ET1*A11*A22-ET1*A12*A21+ET2*A22-ET2*A11*A21+ET2*A12-ANU
     +1*(-B11*A21+B22+B12-B21*A11)*(1.0-G)-ANU2*(B11+B21)*(1.0-G))
      A2=(-G*(A12*A22-ET1*A11*A22-ET1*A12*A21+ET2*A22-ET2*A11*A21+ET2*A1
     +2)-ET1*A12*A22-ET2*A11*A22-ET2*A12*A21+ANU1*(B11*A22+B12*A21+B21*A
     +12+B22*A11)*(1.0-G)-ANU2*(-B11*A21+B22+B12-B21*A11)*(1.0-G))
      A1=(G*(ET1*A12*A22+ET2*A11*A22+ET2*A12*A21)-ET2*A12*A22+ANU2*(B11*
     +A22+B12*A21+B21*A12+B22*A11)*(1.0-G)+ANU1*(B12*A22+B22*A12)*(1.0-
     +G))
      A0=G*ET2*A12*A22+ANU2*(B12*A22+B22*A12)*(1.0-G)
      WRITE(6,60)B5,B4,B3,B2,B1,B0
   60 FORMAT(1X,'B5=',F9.6,2X,'B4=',F9.6,2X,'B3=',F9.6,2X,'B2=',F9.6,2X,
     +'B1=',F9.6,2X,'B0=',F9.6,2X)
      WRITE(6,70)A7,A6,A5,A4,A3,A2,A1,A0
   70 FORMAT(1X,'A7=',F9.6,2X,'A6=',F9.6,2X,'A5=',F9.6,2X,'A4=',F9.6,2X,
     +'A3=',F9.6,2X,'A2=',F9.6,2X,'A1=',F9.6,2X,'A0=',F9.6,2X)
      WRITE(6,30)ET1,ET2,ANU1,ANU2,GM1,GM2
   30 FORMAT(1X,'ET1=',F9.6,2X,'ET2=',F9.6,2X,'NU1=',F9.6,2X,'NU2=',F9.6
     +,2X,'GM1=',F9.6,2X,'GM2=',F9.6,2X)
      WRITE(6,80)
   80 FORMAT(1X,'OVERALL TRANSFER FUNCTION POLES AND ZEROS')
      WRITE(6,65)
   65 FORMAT(//,20X,'***POLES***',40X,'***ZEROS***')
      WRITE(6,45)
   45 FORMAT(/,'            REAL',9X,'IMAG',9X,'MAG.',9X,'ANG.',9X,'REAL',9X
     +,'IMAG',9X,'MAG.',9X,'ANG.')
      CALL MUH1(Z,A7,A6,A5,A4,A3,A2,A1,A0)
      CALL MUH2(Z1,B5,B4,B3,B2,B1,B0)
      DO 66 I =1,7
      ZR=REAL(Z(I))
      ZI=AIMAG(Z(I))
      ZMAG=SQRT(ZR**2+ZI**2)
      IF(ZI.NE.0.0.AND.ZR.NE.0.0)GO TO 12
      IF(ZI.EQ.0.0.AND.ZR.EQ.0.0)ZANG=0.0
      IF(ZI.EQ.0.0.AND.ZR.GT.0.0)ZANG=0.0
      IF(ZI.EQ.0.0.AND.ZR.LT.0.0)ZANG=180.0
```

```
      IF(ZR.EQ.0.0.AND.ZI.GT.0.0)ZANG=90.0
      IF(ZR.EQ.0.0.AND.ZI.LT.0.0)ZANG=270.0
      GO TO 13
12    IF(ZR.LT.0.0)ZANG=(ATAN(ZI/ZR))*(180.0/PI)+180.0
      IF(ZR.GT.0.0)ZANG=(ATAN(ZI/ZR))*(180.0/PI)
13    IF(I.GE.6)GO TO 15
      ZI1=AIMAG(Z1(I))
      ZR1=REAL(Z1(I))
      ZMAG1=SQRT(ZR1**2+ZI1**2)
      IF(ZI1.NE.0.0.AND.ZR1.NE.0.0)GO TO 14
      IF(ZI1.EQ.0.0.AND.ZR1.EQ.0.0)ZANG1=0.0
      IF(ZI1.EQ.0.0.AND.ZR1.GT.0.0)ZANG1=0.0
      IF(ZI1.EQ.0.0.AND.ZR1.LT.0.0)ZANG1=180.0
      IF(ZR1.EQ.0.0.AND.ZI1.GT.0.0)ZANG1=90.0
      IF(ZR1.EQ.0.0.AND.ZI1.LT.0.0)ZANG1=270.0
      GO TO 17
14    IF(ZR1.LT.0.0)ZANG1=(ATAN(ZI1/ZR1))*(180.0/PI)+180.0
      IF(ZR1.GT.0.0)ZANG1=(ATAN(ZI1/ZR1))*(180.0/PI)
17    WRITE(6,65)ZR,ZI,ZMAG,ZANG,ZR1,ZI1,ZMAG1,ZANG1
65    FORMAT(1X,8F13.6)
      GO TO 66
15    WRITE(6,50)ZR,ZI,ZMAG,ZANG
50    FORMAT(1X,4F13.6)
66    CONTINUE
      STOP
      END


      SUBROUTINE MOM1 (Z,A7,A6,A5,A4,A3,A2,A1,A0)
      INTEGER NDEG,IER
      REAL P(8)
      COMPLEX Z(7)
      NDEG=7
      P(1)=A7
      P(2)=A6
      P(3)=A5
      P(4)=A4
      P(5)=A3
      P(6)=A2
      P(7)=A1
      P(8)=A0
      CALL ZRPOLY (P,NDEG,Z,IER)
      RETURN
      END


      SUBROUTINE MOM2 (Z1,B5,B4,B3,B2,B1,B0)
      INTEGER NDEG,IER
      REAL Q(6)
      COMPLEX Z1(5 )
      NDEG=5
      Q(1)=B5
      Q(2)=B4
      Q(3)=B3
      Q(4)=B2
      Q(5)=B1
      Q(6)=B0
      CALL ZRPOLY (Q,NDEG,Z1,IER)
      RETURN
      END
```

B5= 0.037444  B4= 0.046736  B3= 0.044386  B2= 0.032662  B1= 0.0  B0= 0.0
A7= 1.000000  A6=-0.770061  A5=-0.882752  A4= 0.157761  A3= 0.990814  A2=-0.325873  A1=-0.025889
ET1=-0.158567  ET2=-0.071468  NU1=-0.475215  NU2= 0.555756  GM1= 1.687101  GM2=-0.740818  A0=-0.071794
OVERALL TRANSFER FUNCTION POLES AND ZEROS

$\lambda_2 = 1$

| ***POLES*** | | | | ***ZEROS*** | | | |
|---|---|---|---|---|---|---|---|
| REAL | IMAG | MAG. | ANG. | REAL | IMAG | MAG. | ANG. |
| -0.116420 | 0.323645 | 0.343947 | 109.784531 | 0.0 | 0.0 | 0.0 | 0.0 |
| -0.116420 | -0.323645 | 0.343947 | 250.215469 | 0.0 | 0.0 | 0.0 | 0.0 |
| -0.767525 | 0.559482 | 0.949797 | 143.910065 | -0.144124 | 0.942301 | 0.953259 | 98.695969 |
| -0.767525 | -0.559482 | 0.949797 | 216.089935 | -0.144124 | -0.942301 | 0.953259 | 261.303955 |
| 0.844069 | 0.281373 | 0.889732 | 18.435926 | -0.959908 | 0.0 | 0.959908 | 180.000000 |
| 0.844069 | -0.281373 | 0.889732 | -18.435926 | | | | |
| 0.849812 | 0.0 | 0.849812 | 0.0 | | | | |

B5= 0.161415  B4=-0.063227  B3=-0.062788  B2= 0.140197  B1= 0.0  B0= 0.0
A7= 1.000000  A6=-0.770061  A5=-0.882752  A4= 0.275586  A3= 0.748507  A2=-0.305510  A1= 0.195442
ET1=-0.158567  ET2=-0.071468  NU1=-0.475215  NU2= 0.555756  GM1= 1.687101  GM2=-0.740818  A0=-0.191321
OVERALL TRANSFER FUNCTION POLES AND ZEROS

$\lambda_2 = 10$

| ***POLES*** | | | | ***ZEROS*** | | | |
|---|---|---|---|---|---|---|---|
| REAL | IMAG | MAG. | ANG. | REAL | IMAG | MAG. | ANG. |
| -0.069880 | 0.539954 | 0.544457 | 97.374191 | 0.0 | 0.0 | 0.0 | 0.0 |
| -0.069880 | -0.539954 | 0.544457 | 262.625732 | 0.0 | 0.0 | 0.0 | 0.0 |
| -0.828968 | 0.492095 | 0.964025 | 149.305634 | 0.675252 | 0.670752 | 0.951774 | 44.808441 |
| -0.828968 | -0.492095 | 0.964025 | 210.694366 | 0.675252 | -0.670752 | 0.951774 | -44.808441 |
| 0.851238 | 0.279411 | 0.895922 | 18.171951 | -0.958798 | 0.0 | 0.958798 | 180.000000 |
| 0.851238 | -0.279411 | 0.895922 | -18.171951 | | | | |
| 0.865280 | 0.0 | 0.865280 | 0.0 | | | | |

B5= 0.299160  B4=-0.185409  B3=-0.181872  B2= 0.259680  B1= 0.0  B0= 0.0
A7= 1.000000  A6=-0.770061  A5=-0.682752  A4= 0.406503  A3= 0.479276  A2=-0.282884  A1= 0.441365
ET1=-0.158567  ET2=-0.071466  NU1=-0.475215  NU2= 0.555756  GM1= 1.687101  GM2=-0.740818  A0=-0.324128
OVERALL TRANSFER FUNCTION POLES AND ZEROS

$\lambda_2 = 20$

| ***POLES*** | | | | ***ZEROS*** | | | |
|---|---|---|---|---|---|---|---|
| REAL | IMAG | MAG. | ANG. | REAL | IMAG | MAG. | ANG. |
| -0.022221 | 0.675106 | 0.675471 | 91.885239 | 0.0 | 0.0 | 0.0 | 0.0 |
| -0.022221 | -0.675106 | 0.675471 | 268.114746 | 0.0 | 0.0 | 0.0 | 0.0 |
| -0.889974 | 0.449986 | 0.997267 | 153.178055 | 0.789246 | 0.531500 | 0.951526 | 33.957397 |
| -0.889974 | -0.449986 | 0.997267 | 206.821945 | 0.789246 | -0.531500 | 0.951526 | -33.957397 |
| 0.858554 | 0.277574 | 0.902310 | 17.916214 | -0.958726 | 0.0 | 0.958726 | 180.000000 |
| 0.858554 | -0.277574 | 0.902310 | -17.916214 | | | | |
| 0.877340 | 0.0 | 0.877340 | 0.0 | | | | |

B5= 0.712395  B4=-0.551954  B3=-0.539122  B2= 0.618131  B1= 0.0  B0= 0.0
A7= 1.000000  A6=-0.770061  A5=-0.682752  A4= 0.799254  A3=-0.328415  A2=-0.215006  A1= 1.179135
ET1=-0.158567  ET2=-0.071468  NU1=-0.475215  NU2= 0.555756  GM1= 1.687101  GM2=-0.740818  A0=-0.722550
OVERALL TRANSFER FUNCTION POLES AND ZEROS

$\lambda_2 = 50$

| ***POLES*** | | | | ***ZEROS*** | | | |
|---|---|---|---|---|---|---|---|
| REAL | IMAG | MAG. | ANG. | REAL | IMAG | MAG. | ANG. |
| 0.906295 | 0.0 | 0.906295 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| -1.013580 | 0.373931 | 1.080356 | 159.749924 | 0.0 | 0.0 | 0.0 | 0.0 |
| -1.013580 | -0.373931 | 1.080356 | 200.250076 | 0.866744 | 0.392230 | 0.951362 | 24.348297 |
| 0.072635 | 0.900785 | 0.903709 | 85.389847 | 0.866744 | -0.392230 | 0.951362 | -24.348297 |
| 0.072635 | -0.900785 | 0.903709 | -85.389847 | -0.958700 | 0.0 | 0.958700 | 180.000000 |
| 0.872828 | 0.273215 | 0.914590 | 17.381302 | | | | |
| 0.872828 | -0.273215 | 0.914590 | -17.381302 | | | | |

```
C     THIS PROGRAM DETERMINES CONTROLLED SYSTEM PARS. AND SINGS.
      DIMENSION G1(2),F1(2),ZET(2),W(2),AL(2),G2(2),F2(2)
      COMPLEX Z(7),Z1(5)
      DATA T/0.5/,AL/1.0,1.0/,ZET/0.2,0.02/,W/0.5,5.0/,D/0.5/,ZETA/0.6/
     +,SIG/1.0/,PI/3.141592/
      DO 10 I=1,2
      G1(I)=2*EXP(-ZET(I)*W(I)*T)*COS(W(I)*T*SQRT(1-ZET(I)**2))
      IF(I.EQ.1)A11=G1(1)
      IF(I.EQ.2)A21=G1(2)
      G2(I)=-EXP(-2*ZET(I)*W(I)*T)
      IF(I.EQ.1)A12=G2(1)
      IF(I.EQ.2)A22=G2(2)
      F1(I)=(AL(I)/W(I)**2)*(1-EXP(-ZET(I)*W(I)*T)*(COS(W(I)*T*SQRT(1-ZE
     +T(I)**2))+(ZET(I)/SQRT(1-ZET(I)**2))*SIN(W(I)*T*SQRT(1-ZET(I)**2))
     +))
      IF(I.EQ.1)B11=F1(1)
      IF(I.EQ.2)B21=F1(2)
      F2(I)=(AL(I)/W(I)**2)*(EXP(-ZET(I)*W(I)*T)*(EXP(-ZET(I)*W(I)*T)-CO
     +S(W(I)*T*SQRT(1-ZET(I)**2))+(ZET(I)/SQRT(1-ZET(I)**2))*SIN(W(I)*T*
     +SQRT(1-ZET(I)**2))))
      IF(I.EQ.1)B12=F2(1)
      IF(I.EQ.2)B22=F2(2)
10    CONTINUE
      GM1=2*EXP(-ZETA*W(1)*T)*COS(W(1)*T*SQRT(1.0-ZETA**2))
      GM2=-EXP(-2*ZETA*W(1)*T)
      ET1=GM1-A11
      ANU1=(ET1*A12+(ET1*A11-A12+GM2)*(A11-A12*B11/B12))/(B11*A11+B12-A1
     +2*B11**2/B12)
      ET2=A11*ET1+GM2-B11*ANU1-A12
      ANU2=ET2*A12/B12
      G=EXP(-SIG*T)
      B5=D*(1.0-G)*(B11+B21)
      B4=D*(1.0-G)*(-B11*A21+B12-B21*A11+B22)
      B3=D*(1.0-G)*(-B11*A22-B12*A21-B21*A12-B22*A11)
      B2=D*(1.0-G)*(-B12*A22-B22*A12)
      B1=0.0
      B0=B1
      A7=1.0
      A6=-A21-A11-G
      A5=-A22+A11*A21-A12+G*(A21+A11)-ET1*(1.0-G)
      A4=A11*A22+A12*A21-G*(-A22+A11*A21-A12)+ET1*(1.0-G)*(A21+A11)-ET2*
     +(1.0-G)-ANU1*(1.0-G)*(B11+B21)
      A3=A12*A22-G*(A11*A22+A12*A21)-ET1*(1.0-G)*(-A22+A11*A21-A12)-ET2*
     +(1.0-G)*(-A21-A11)-ANU1*(1.0-G)*(-B11*A21+B22+B12-B21*A11)-ANU2*(1
     +.0-G)*(B11+B21)
      A2=-G*A12*A22-ET1*(1.0-G)*(A11*A22+A12*A21)-ET2*(1.0-G)*(-A22+A11*
     +A21-A12)+ANU1*(1.0-G)*(B11*A22+B12*A21+B21*A12+B22*A11)-ANU2*(1.0-
     +G)*(-B11*A21+B22+B12-B21*A11)
      A1=-ET1*(1.0-G)*A12*A22-ET2*(1.0-G)*(A11*A22+A12*A21)+ANU2*(1.0-G)
     +*(B11*A22+B12*A21+B21*A12+B22*A11)+ANU1*(1.0-G)*(B12*A22+B22*A12)
      A0=-ET2*(1.0-G)*A12*A22+ANU2*(1.0-G)*(B12*A22+B22*A12)
      WRITE(6,60)B5,B4,B3,B2,B1,B0
60    FORMAT(1X,'B5=',F9.6,2X,'B4=',F9.6,2X,'B3=',F9.6,2X,'B2=',F9.6,2X,
     +'B1=',F9.6,2X,'B0=',F9.6,2X)
      WRITE(6,70)A7,A6,A5,A4,A3,A2,A1,A0
70    FORMAT(1X,'A7=',F9.6,2X,'A6=',F9.6,2X,'A5=',F9.6,2X,'A4=',F9.6,2X,
     +'A3=',F9.6,2X,'A2=',F9.6,2X,'A1=',F9.6,2X,'A0=',F9.6,2X)
      WRITE(6,30)ET1,ET2,ANU1,ANU2,GM1,GM2
30    FORMAT(1X,'ET1=',F9.6,2X,'ET2=',F9.6,2X,'NU1=',F9.6,2X,'NU2=',F9.6
     +,2X,'GM1=',F9.6,2X,'GM2=',F9.6,2X)
      WRITE(6,80)
80    FORMAT(1X,'OVERALL TRANSFER FUNCTION POLES AND ZEROS')
      WRITE(6,85)
85    FORMAT(//,20X,'***POLES***',40X,'***ZEROS***')
      WRITE(6,45)
45    FORMAT(/,'          REAL',9X,'IMAG',9X,'MAG.',9X,'ANG.',9X,'REAL',9X
     +,'IMAG',9X,'MAG.',9X,'ANG.')
      CALL MOH1(Z,A7,A6,A5,A4,A3,A2,A1,A0)
      CALL MOH2(Z1,B5,B4,B3,B2,B1,B0)
      DO 66 I =1,7
      ZR=REAL(Z(I))
      ZI=AIMAG(Z(I))
      ZMAG=SQRT(ZR**2+ZI**2)
      IF(ZI.NE.0.0.AND.ZR.NE.0.0)GO TO 12
      IF(ZI.EQ.0.0.AND.ZR.EQ.0.0)ZANG=0.0
      IF(ZI.EQ.0.0.AND.ZR.GT.0.0)ZANG=0.0
      IF(ZI.EQ.0.0.AND.ZR.LT.0.0)ZANG=180.0
```

```
      IF(ZR.EQ.0.0.AND.ZI.GT.0.0)ZANG=90.0
      IF(ZR.EQ.0.0.AND.ZI.LT.0.0)ZANG=270.0
      GO TO 13
12    IF(ZR.LT.0.0)ZANG=(ATAN(ZI/ZR))*(180.0/PI)+180.0
      IF(ZR.GT.0.0)ZANG=(ATAN(ZI/ZR))*(180.0/PI)
13    IF(I.GE.6)GO TO 15
      ZI1=AIMAG(Z1(I))
      ZR1=REAL(Z1(I))
      ZMAG1=SQRT(ZR1**2+ZI1**2)
      IF(ZI1.NE.0.0.AND.ZR1.NE.0.0)GO TO 14
      IF(ZI1.EQ.0.0.AND.ZR1.EQ.0.0)ZANG1=0.0
      IF(ZI1.EQ.0.0.AND.ZR1.GT.0.0)ZANG1=0.0
      IF(ZI1.EQ.0.0.AND.ZR1.LT.0.0)ZANG1=180.0
      IF(ZR1.EQ.0.0.AND.ZI1.GT.0.0)ZANG1=90.0
      IF(ZR1.EQ.0.0.AND.ZI1.LT.0.0)ZANG1=270.0
      GO TO 17
14    IF(ZR1.LT.0.0)ZANG1=(ATAN(ZI1/ZR1))*(180.0/PI)+180.0
      IF(ZR1.GT.0.0)ZANG1=(ATAN(ZI1/ZR1))*(180.0/PI)
17    WRITE(6,65)ZR,ZI,ZMAG,ZANG,ZR1,ZI1,ZMAG1,ZANG1
65    FORMAT(1X,8F13.6)
      GO TO 66
15    WRITE(6,50)ZR,ZI,ZMAG,ZANG
50    FORMAT(1X,4F13.6)
66    CONTINUE
      STOP
      END


      SUBROUTINE MOM1 (Z,A7,A6,A5,A4,A3,A2,A1,A0)
      INTEGER NDEG,IER
      REAL P(8)
      COMPLEX Z(7)
      NDEG=7
      P(1)=A7
      P(2)=A6
      P(3)=A5
      P(4)=A4
      P(5)=A3
      P(6)=A2
      P(7)=A1
      P(8)=A0
      CALL ZRPOLY (P,NDEG,Z,IER)
      RETURN
      END


      SUBROUTINE MOM2 (Z1,B5,B4,B3,B2,B1,B0)
      INTEGER NDEG,IER
      REAL Q(6)
      COMPLEX Z1(5 )
      NDEG=5
      Q(1)=B5
      Q(2)=B4
      Q(3)=B3
      Q(4)=B2
      Q(5)=B1
      Q(6)=B0
      CALL ZRPOLY (Q,NDEG,Z1,IER)
      RETURN
      END
```

B5= 0.037444 B4= 0.046736 B3= 0.044386 B2= 0.032862 B1= 0.0 B0= 0.0
A7= 1.000000 A6=-0.928628 A5=-0.744578 A4= 0.360112 A3= 0.926707 A2=-0.552717 A1= 0.024593
ET1=-0.158567 ET2=-0.071468 NU1=-0.475215 NU2= 0.555756 GM1= 1.687101 GM2=-0.740818 A0=-0.013281
OVERALL TRANSFER FUNCTION POLES AND ZEROS

$\lambda_2 = 1$

| ***POLES*** | | | | ***ZEROS*** | | | |
|---|---|---|---|---|---|---|---|
| REAL | IMAG | MAG. | ANG. | REAL | IMAG | MAG. | ANG. |
| 0.001859 | 0.154233 | 0.154244 | 89.309494 | 0.0 | 0.0 | 0.0 | 0.0 |
| 0.001859 | -0.154233 | 0.154244 | -89.309494 | 0.0 | 0.0 | 0.0 | 0.0 |
| -0.767098 | 0.561032 | 0.950366 | 143.819427 | -0.144124 | 0.942301 | 0.953259 | 98.695969 |
| -0.767098 | -0.561032 | 0.950366 | 216.180573 | -0.144124 | -0.942301 | 0.953259 | 261.303955 |
| 0.819335 | 0.286425 | 0.867957 | 19.268692 | -0.959908 | 0.0 | 0.959908 | 180.000000 |
| 0.819335 | -0.286425 | 0.867957 | -19.268692 | | | | |
| 0.820436 | 0.0 | 0.820436 | 0.0 | | | | |

B5= 0.161415 B4=-0.083227 B3=-0.062788 B2= 0.140197 B1= 0.0 B0= 0.0
A7= 1.000000 A6=-0.928628 A5=-0.744578 A4= 0.477937 A3= 0.664400 A2=-0.532354 A1= 0.245924
ET1=-0.158567 ET2=-0.071468 NU1=-0.475215 NU2= 0.555756 GM1= 1.687101 GM2=-0.740818 A0=-0.132807
OVERALL TRANSFER FUNCTION POLES AND ZEROS

$\lambda_2 = 10$

| ***POLES*** | | | | ***ZEROS*** | | | |
|---|---|---|---|---|---|---|---|
| REAL | IMAG | MAG. | ANG. | REAL | IMAG | MAG. | ANG. |
| 0.031599 | 0.467446 | 0.468513 | 86.132690 | 0.0 | 0.0 | 0.0 | 0.0 |
| 0.031599 | -0.467446 | 0.468513 | -86.132690 | 0.0 | 0.0 | 0.0 | 0.0 |
| -0.821880 | 0.501240 | 0.962667 | 148.622238 | 0.675252 | 0.670752 | 0.951774 | 44.808441 |
| -0.821880 | -0.501240 | 0.962667 | 211.377762 | 0.675252 | -0.670752 | 0.951774 | -44.808441 |
| 0.834234 | 0.283548 | 0.881105 | 18.772369 | -0.958796 | 0.0 | 0.958796 | 180.000000 |
| 0.834234 | -0.283548 | 0.881105 | -18.772369 | | | | |
| 0.840722 | 0.0 | 0.840722 | 0.0 | | | | |

B5= 0.299166 B4=-0.185409 B3=-0.161872 B2= 0.259666 B1= 0.0 B0= 0.0
A7= 1.000000 A6=-0.928628 A5=-0.744578 A4= 0.606854 A3= 0.415169 A2=-0.509727 A1= 0.491847
ET1=-0.158567 ET2=-0.071468 NU1=-0.475215 NU2= 0.555756 GM1= 1.657101 GM2=-0.740818 A0=-0.265614
OVERALL TRANSFER FUNCTION POLES AND ZEROS

$\lambda_2 = 20$

| ***POLES*** | | | | ***ZEROS*** | | | |
|---|---|---|---|---|---|---|---|
| REAL | IMAG | MAG. | ANG. | REAL | IMAG | MAG. | ANG. |
| 0.067908 | 0.625655 | 0.629329 | 83.805298 | 0.0 | 0.0 | 0.0 | 0.0 |
| 0.067908 | -0.625655 | 0.629329 | -83.805298 | 0.0 | 0.0 | 0.0 | 0.0 |
| -0.878186 | 0.461232 | 0.991940 | 152.291107 | 0.789240 | 0.531500 | 0.951526 | 33.957397 |
| -0.878186 | -0.461232 | 0.991940 | 207.708893 | 0.789240 | -0.531500 | 0.951526 | -33.957397 |
| 0.843529 | 0.281092 | 0.889131 | 18.429816 | -0.958720 | 0.0 | 0.958720 | 180.000000 |
| 0.843529 | -0.281092 | 0.889131 | -18.429816 | | | | |
| 0.862125 | 0.0 | 0.862125 | 0.0 | | | | |

B5= 0.712595 B4=-0.551954 B3=-0.539122 B2= 0.818131 B1= 0.0 B0= 0.0
A7= 1.000000 A6=-0.928628 A5=-0.744578 A4= 1.001005 A3=-0.392522 A2=-0.441850 A1= 1.229617
ET1=-0.158567 ET2=-0.071465 NU1=-0.475215 NU2= 0.555756 GM1= 1.687101 GM2=-0.740818 A0=-0.664037
OVERALL TRANSFER FUNCTION POLES AND ZEROS

$\lambda_2 = 50$

| ***POLES*** | | | | ***ZEROS*** | | | |
|---|---|---|---|---|---|---|---|
| REAL | IMAG | MAG. | ANG. | REAL | IMAG | MAG. | ANG. |
| 0.864092 | 0.276116 | 0.907136 | 17.721008 | 0.0 | 0.0 | 0.0 | 0.0 |
| 0.864092 | -0.276116 | 0.907136 | -17.721008 | 0.0 | 0.0 | 0.0 | 0.0 |
| -0.996985 | 0.389109 | 1.070249 | 158.677017 | 0.866744 | 0.392250 | 0.951362 | 24.346297 |
| -0.996985 | -0.389109 | 1.070249 | 201.322983 | 0.866744 | -0.392250 | 0.951362 | -24.346297 |
| 0.148511 | 0.873011 | 0.885553 | 80.345551 | -0.958700 | 0.0 | 0.958700 | 180.000000 |
| 0.148511 | -0.873011 | 0.885553 | -80.345551 | | | | |
| 0.897591 | 0.0 | 0.897591 | 0.0 | | | | |

## Appendix F: Sample Programs for the Five Adaptive Controllers in Section 4 and the Run Statistics in Section 5.

```
      BLOCK DATA
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 IDO(10),IO(10),NITE(10),SASI(10),SAPPEA(10),
     *  SAPPEB(10),SPPEVA(10),SPPEVB(10),SACPEF(10),SACPEG(10),
     *  SCPEVF(10),SCPEVG(10),NASTE(10),NSTEV(10)
      INTEGER KD(12)
      COMMON /STATS/IDO,IO,NITE,SASI,SAPPEA,SAPPEB,
     *  SPPEVA,SPPEVB,SACPEF,SACPEG,SCPEVF,SCPEVG,NASTE,NSTEV,
     *  STE,SSPPA,SSPPB,SSCPF,SSCPG,M,KD
      DATA KD/0,1,2,5,10,20,50,100,200,500,1000,2000/
      END

      IMPLICIT REAL*8 (A-H,O-Z)
      EXTERNAL OVFL
      REAL*8 A1(5),A2(5),B(5),E(5),AMP(4),W(4,5)
      REAL*8 C(5),D(5),AHAT(4),BHAT(4),FHAT(4),GHAT(4)
      REAL YMIN(4),YMAX(4)
      REAL*8 IDO(10),IO(10),NITE(10),SASI(10),SAPPEA(10),
     *  SAPPEB(10),SPPEVA(10),SPPEVB(10),SACPEF(10),SACPEG(10),
     *  SCPEVF(10),SCPEVG(10),NASTE(10),NSTEV(10)
      REAL*8 ESTMUL(3)
      INTEGER PCTL,PTEMP,PXCHG,ESTSTR(3)
      INTEGER KD(12)
      COMMON /STATS/IDO,IO,NITE,SASI,SAPPEA,SAPPEB,
     *  SPPEVA,SPPEVB,SACPEF,SACPEG,SCPEVF,SCPEVG,NASTE,NSTEV,
     *  STE,SSPPA,SSPPB,SSCPF,SSCPG,M,KD
      COMMON /LUN/LP
      COMMON /CONTRL/F,G,AHAT,BHAT,FHAT,GHAT
      DATA K/4/
      DATA A1/0.95D0,0.9D0,0.9D0,0.9D0,0.83D0/
      DATA A2/0.2D0,0.1D0,0.8D0,0.7D0,0.3D0/
      DATA B/0.065D0,0.1D0,0.08D0,0.5D0,-0.193D0/
      DATA E/0.01D0,-0.01D0,0.02D0,-0.3D0,0.208D0/
      DATA C/0.4D0,0.7D0,1.0D0,0.8D0,0.6D0/
      DATA D/0.8D0,0.65D0,0.5D0,0.6D0,0.7D0/
      DATA YMIN/-1.0,-4.0,-4.0,-1.0/
      DATA YMAX/2.0,4.0,4.0,2.0/
      DATA MAX,NSEED/1000,13579/
      DATA AMP/1.0D0,2.0D0,2.0D0,2.0D0/
      DATA W(1,1),W(1,2),W(1,3),W(1,4),W(1,5)/
     *  0.17851D0,0.34463D0,0.55452D0,0.40866D0,0.28534D0/
      DATA W(2,1),W(2,2),W(2,3),W(2,4),W(2,5)/
     *  0.44629D0,0.86157D0,1.38629D0,1.0217D0,0.71335D0/
      DATA W(3,1),W(3,2),W(3,3),W(3,4),W(3,5)/
     *  0.17851D0,0.34463D0,0.55452D0,0.40866D0,0.28534D0/
      DATA W(4,1),W(4,2),W(4,3),W(4,4),W(4,5)/
     *  0.44629D0,0.86157D0,1.38629D0,1.0217D0,0.71335D0/
      DATA ESTMUL/0.8D0,1.0D0,1.2D0/
      DATA PCTL,PTEMP/'1','0'/
      DATA ESTSTR/'-20%',' 0  ','+20%'/

      LP = 6
      CALL ERRSET(207,1000,-1,1,OVFL,0)
      WRITE(6,10)
   10 FORMAT(' PRINTOUT INTERVAL = ')
```

```
      READ(5,*)INTV
      WRITE(6,20)
   20 FORMAT(' STARTING AND ENDING EXAMPLES = ')
      READ(5,*)NEXST,NEXEND
      WRITE(6,30)
   30 FORMAT(' STARTING AND ENDING ALGORITHMS = ')
      READ(5,*)NALGST,NALGED
      NSC = 2
      DO 1000 NEX=NEXST,NEXEND
      DO 100 NALG=NALGST,NALGED
      DO 100 NEST=1,3
      DO 100 INP=1,4
      AHAT(K)  = A1(NEX)*ESTMUL(NEST)
      AHAT(K-1)  = AHAT(K)
      BHAT(K)  = B(NEX)*ESTMUL(NEST)
      BHAT(K-1)  = BHAT(K)
      GHAT(K-1)  = C(NEX)/B(NEX)
      GHAT(K-1)  = GHAT(K-1)
      GHAT(K-2)  = GHAT(K-1)
      G = GHAT(K-1)
      FHAT(K-1)  =  (D(NEX)-A1(NEX))/B(NEX)
      FHAT(K-1)  = FHAT(K-1)
      FHAT(K-2)  = FHAT(K-1)
      F = FHAT(K-1)
      WRITE(10,50)PCTL,NEX,INP,ESTSTR(NEST)
   50 FORMAT(A1,15X,'EXAMPLE',I2,'   INPUT',I2,'  NON-ADAPTIVE',
     *  '   EST = ',A4/)
      WRITE(6,50)PCTL,NEX,INP,ESTSTR(NEST)
      PXCHG = PCTL
      PCTL = PTEMP
      PTEMP = PXCHG
      CALL SIMUL(A1(NEX),A2(NEX),B(NEX),E(NEX),AMP(INP),W(INP,NEX),
     *  NSEED,INP,NALG,NEX,NSC,C(NEX),D(NEX),YMIN(INP),YMAX(INP),
     *  INTV,MAX)
  100 CONTINUE
 1000 CONTINUE
      ENDFILE 10
      STOP
      END
      SUBROUTINE OVFL
      LOGICAL OVFLOW
      COMMON /BOMB/OVFLOW
      WRITE(6,10)
   10 FORMAT(/' *** OVERFLOW ***'/)
      OVFLOW = .TRUE.
      RETURN
      END
```

```
      BLOCK DATA
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 IDO(10),IO(10),NITE(10),SASI(10),SAPPEA(10),
     * SAPPEB(10),SPPEVA(10),SPPEVB(10),SACPEF(10),SACPEG(10),
     * SCPEVF(10),SCPEVG(10),NASTE(10),NSTEV(10)
      INTEGER KD(12)
      COMMON /STATS/IDO,IO,NITE,SASI,SAPPEA,SAPPEB,
     * SPPEVA,SPPEVB,SACPEF,SACPEG,SCPEVF,SCPEVG,NASTE,NSTEV,
     * STE,SSPPA,SSPPB,SSCPF,SSCPG,M,KD
      DATA KD/0,1,2,5,10,20,50,100,200,500,1000,2000/
      END

      IMPLICIT REAL*8 (A-H,O-Z)
      EXTERNAL OVFL
      REAL*8 A1(5),A2(5),B(5),E(5),AMP(4),W(4,5)
      REAL*8 C(5),D(5),AHAT(4),BHAT(4),FHAT(4),GHAT(4)
      REAL YMIN(4),YMAX(4)
      REAL*8 IDO(10),IO(10),NITE(10),SASI(10),SAPPEA(10),
     * SAPPEB(10),SPPEVA(10),SPPEVB(10),SACPEF(10),SACPEG(10),
     * SCPEVF(10),SCPEVG(10),NASTE(10),NSTEV(10)
      REAL*8 ESTMUL(3)
      INTEGER PCTL,PTEMP,PXCHG,ESTSTR(3),SCSTR(3)
      INTEGER KD(12)
      COMMON /STATS/IDO,IO,NITE,SASI,SAPPEA,SAPPEB,
     * SPPEVA,SPPEVB,SACPEF,SACPEG,SCPEVF,SCPEVG,NASTE,NSTEV,
     * STE,SSPPA,SSPPB,SSCPF,SSCPG,M,KD
      COMMON /LUN/LP
      COMMON /CONTRL/F,G,AHAT,BHAT,FHAT,GHAT
      DATA K/4/
      DATA A1/0.95D0,0.9D0,0.9D0,0.9D0,0.83D0/
      DATA A2/0.2D0,0.1D0,0.8D0,0.7D0,0.3D0/
      DATA B/0.065D0,0.1D0,0.08D0,0.5D0,-0.193D0/
      DATA E/0.01D0,-0.01D0,0.02D0,-0.3D0,0.208D0/
      DATA C/0.4D0,0.7D0,1.0D0,0.8D0,0.6D0/
      DATA D/0.8D0,0.65D0,0.5D0,0.6D0,0.7D0/
      DATA YMIN/-1.0,-4.0,-4.0,-1.0/
      DATA YMAX/2.0,4.0,4.0,2.0/
      DATA MAX,NSEED/1000,13579/
      DATA AMP/1.0D0,2.0D0,2.0D0,2.0D0/
      DATA W(1,1),W(1,2),W(1,3),W(1,4),W(1,5)/
     * 0.17851D0,0.34463D0,0.55452D0,0.40866D0,0.28534D0/
      DATA W(2,1),W(2,2),W(2,3),W(2,4),W(2,5)/
     * 0.44629D0,0.86157D0,1.38629D0,1.0217D0,0.71335D0/
      DATA W(3,1),W(3,2),W(3,3),W(3,4),W(3,5)/
     * 0.17851D0,0.34463D0,0.55452D0,0.40866D0,0.28534D0/
      DATA W(4,1),W(4,2),W(4,3),W(4,4),W(4,5)/
     * 0.44629D0,0.86157D0,1.38629D0,1.0217D0,0.71335D0/
      DATA ESTMUL/0.8D0,1.0D0,1.2D0/
      DATA PCTL,PTEMP/'1','0'/
      DATA ESTSTR/'-20%',' 0  ','+20%'/
      DATA SCSTR/'SC- ','SC0 ','SC+ '/

      LP = 6
      CALL ERRSET(207,1000,-1,1,OVFL,0)
      WRITE(6,10)
```

```
 10 FORMAT(' PRINTOUT INTERVAL = ')
    READ(5,*)INTV

    WRITE(6,20)
 20 FORMAT(' STARTING AND ENDING EXAMPLES = ')
    READ(5,*)NEXST,NEXEND
    WRITE(6,30)
 30 FORMAT(' STARTING AND ENDING ALGORITHMS = ')
    READ(5,*)NALGST,NALGED
    WRITE(6,40)
 40 FORMAT(' SMOOTHING COEFFICIENT = ')
    READ(5,*)NSC
    DO 1000 NEX=NEXST,NEXEND
    DO 100 NALG=NALGST,NALGED
    DO 100 NEST=1,3
    DO 100 INP=1,4
    AHAT(K)   = A1(NEX)*ESTMUL(NEST)
    AHAT(K-1) = AHAT(K)
    BHAT(K)   = B(NEX)*ESTMUL(NEST)
    BHAT(K-1) = BHAT(K)
    GHAT(K-1) = C(NEX)/B(NEX)
    GHAT(K-1) = GHAT(K-1)
    GHAT(K-2) = GHAT(K-1)
    G = GHAT(K-1)
    FHAT(K-1) = (D(NEX)-A1(NEX))/B(NEX)
    FHAT(K-1) = FHAT(K-1)
    FHAT(K-2) = FHAT(K-1)
    F = FHAT(K-1)
    WRITE(10,50)PCTL,NEX,INP,NALG,ESTSTR(NEST),SCSTR(NSC)
 50 FORMAT(A1,15X,'EXAMPLE',I2,'    INPUT',I2,'   ALGORITHM',I2,
   *  '    EST = ',A4,'   SS2   ',A4/)
    WRITE(6,50)PCTL,NEX,INP,NALG,ESTSTR(NEST),SCSTR(NSC)
    PXCHG = PCTL
    PCTL = PTEMP
    PTEMP = PXCHG
    CALL SIMUL(A1(NEX),A2(NEX),B(NEX),E(NEX),AMP(INP),W(INP,NEX),
   *  NSEED,INP,NALG,NEX,NSC,C(NEX),D(NEX),YMIN(INP),YMAX(INP),
   *  INTV,MAX)
100 CONTINUE
1000 CONTINUE
    ENDFILE 10
    STOP
    END
    SUBROUTINE OVFL
    LOGICAL OVFLOW
    COMMON /BOMB/OVFLOW
    WRITE(6,10)
 10 FORMAT(/' *** OVERFLOW ***'/)
    OVFLOW = .TRUE.
    RETURN
    END
```

```
      SUBROUTINE SIMUL(A1,A2,B,E,AMP,W,NSEED,
     *  MODE,NALG,NEX,NSC,C,D,YMIN,YMAX,INTV,MAX)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 MU,U(4),Y(4),R(4),S(4)
      REAL*8 AHAT(4),BHAT(4),FHAT(4),GHAT(4)
      REAL SIG(4),TIME
      INTEGER SYMBOL(4)
      REAL*8 IDO(10),IO(10),NITE(10),SASI(10),SAPPEA(10),
     *  SAPPEB(10),SPPEVA(10),SPPEVB(10),SACPEF(10),SACPEG(10),
     *  SCPEVF(10),SCPEVG(10),NASTE(10),NSTEV(10)
      REAL*8 T1(4),T2(4),T3(4),T4(4),T5(4),T6(4),T7(4),
     *  T8(4),T9(4)
      REAL*8 SC(4,3,2)
      INTEGER KD(12)
      LOGICAL OVFLOW
      COMMON /STATS/IDO,IO,NITE,SASI,SAPPEA,SAPPEB,
     *  SPPEVA,SPPEVB,SACPEF,SACPEG,SCPEVF,SCPEVG,NASTE,NSTEV,
     *  STE,SSPPA,SSPPB,SSCPF,SSCPG,M,KD
      COMMON /LUN/LP
      COMMON /BOMB/OVFLOW
      COMMON /SIGNAL/R,U,Y,S
      COMMON /CONTRL/F,G,AHAT,BHAT,FHAT,GHAT
      COMMON /MISC1/T1,T2,T3
      COMMON /MISC2/T4,T5,T6
      COMMON /MISC3/T7,T8,T9
      DATA K/4/
      DATA SC(1,1,1),SC(1,2,1),SC(1,3,1)/-0.9,-0.95,-0.97/
      DATA SC(1,1,2),SC(1,2,2),SC(1,3,2)/-0.7,-0.8,-0.9/
      DATA SC(2,1,1),SC(2,2,1),SC(2,3,1)/-0.9,-0.9,-0.9/
      DATA SC(2,1,2),SC(2,2,2),SC(2,3,2)/-0.65,-0.65,-0.65/
      DATA SC(3,1,1),SC(3,2,1),SC(3,3,1)/-0.9,-0.9,-0.9/
      DATA SC(3,1,2),SC(3,2,2),SC(3,3,2)/-0.5,-0.5,-0.5/
      DATA SC(4,1,1),SC(4,2,1),SC(4,3,1)/-0.8,-0.9,-0.95/
      DATA SC(4,1,2),SC(4,2,2),SC(4,3,2)/-0.4,-0.6,-0.8/
      DATA SYMBOL/'U','R','S','Y'/

      OVFLOW = .FALSE.

      DO 100 I=1,4
      U(I) = 0.0D0
      Y(I) = 0.0D0
      R(I) = 0.0D0
      S(I) = 0.0D0
      T1(I) = 0.0D0
      T2(I) = 0.0D0
      T3(I) = 0.0D0
      T4(I) = 0.0D0
      T5(I) = 0.0D0
      T6(I) = 0.0D0
      T7(I) = 0.0D0
      T8(I) = 0.0D0
      T9(I) = 0.0D0
  100 CONTINUE
      NSD = NSEED
      M = 1
```

```
      DO 200 I=1,10
      IDO(I)  = 0.0D0
      IO(I)   = 0.0D0
      SASI(I) = 0.0D0
      NASTE(I) = 0.0D0
      NSTEV(I) = 0.0D0
      NITE(I) = 0.0D0
      SAPPEA(I) = 0.0D0
      SAPPEB(I) = 0.0D0
      SACPEF(I) = 0.0D0
      SACPEG(I) = 0.0D0
      SPPEVA(I) = 0.0D0
      SPPEVB(I) = 0.0D0
      SCPEVF(I) = 0.0D0
      SCPEVG(I) = 0.0D0
  200 CONTINUE
      KOUNT = 0
      STE = 0.0D0
      SSPPA = 0.0D0
      SSPPB = 0.0D0
      SSCPF = 0.0D0
      SSCPG = 0.0D0
      CY1 = A1+A2
      CY2 = -A1*A2
      CU1 = B+E
      CU2 = -B*A2-E*A1
      IPRT = 0
      INIT = 1
C
      MU = 1.0D0
      RHO = 1.0D0
C
      H = 1.0D0
      IF(NALG.EQ.3) Q = SC(NEX,NSC,1)
      IF(NALG.EQ.5) Q = SC(NEX,NSC,2)
C
      DO 5000 N=1,MAX
C
      IF(OVFLOW) GOTO 6000
C
      Y(K)  = CY1*Y(K-1)+CY2*Y(K-2)+CU1*U(K-1)+CU2*U(K-2)
      S(K)  = C*R(K-1)+D*S(K-1)
C
C        COMMENT OUT THIS COMPUTED GOTO AND ALSO THE
C        FOLLOWING CALL'S FOR THE NON-ADAPTIVE CASE
C
C     GOTO (210,220,230,240,250),NALG
C 210 CALL ADAPT1(C,D,MU,RHO,H)
C     GOTO 300
C 220 CALL ADAPT2(C,D,MU,RHO,H)
C     GOTO 300
C 230 CALL ADAPT3(C,D,MU,RHO,H,Q)
C     GOTO 300
C 240 CALL ADAPT4(C,D,MU,RHO,H)
C     GOTO 300
```

```
 250 CALL ADAPT5(C,D,MU,RHO,H,Q)
 300 CONTINUE

     CALL INP(MODE,N,AMP,W,NSD,R(K))
     U(K) = G*R(K)+F*Y(K)
     TIME = FLOAT(N)

     CALL STAT(N,KOUNT)

     IF((N/INTV)*INTV.NE.N) GOTO 1000
     SIG(1) = SNGL(U(K))
     SIG(2) = SNGL(R(K))
     SIG(3) = SNGL(S(K))
     SIG(4) = SNGL(Y(K))
     CALL TTPLOT(TIME,SIG,4,SYMBOL,YMIN,YMAX,IPRT,INIT)

 000 CONTINUE
     R(K-2) = R(K-1)
     R(K-1) = R(K)
     U(K-2) = U(K-1)
     U(K-1) = U(K)
     Y(K-2) = Y(K-1)
     Y(K-1) = Y(K)
     S(K-2) = S(K-1)
     S(K-1) = S(K)
5000 CONTINUE
     GOTO 6500
6000 CONTINUE
     IDO(M) = 0.0D0
     IO(M)  = 0.0D0
     SASI(M)   = 0.0D0
     NASTE(M) = 0.0D0
     NSTEV(M) = 0.0D0
     NITE(M)  = 0.0D0
     SAPPEA(M) = 0.0D0
     SAPPEB(M) = 0.0D0
     SACPEF(M) = 0.0D0
     SACPEG(M) = 0.0D0
     SPPEVA(M) = 0.0D0
     SPPEVB(M) = 0.0D0
     SCPEVF(M) = 0.0D0
     SCPEVG(M) = 0.0D0
6500 CONTINUE
     WRITE(10,7000) (KD(I),I=2,6)
7000 FORMAT(11X,' K',6X,5(2X,I5,4X)/)
     WRITE(10,7010) (IDO(I),I=1,5)
7010 FORMAT(11X,' IDO',4X,5(1PD11.3))
     WRITE(10,7020) (IO(I),I=1,5)
7020 FORMAT(11X,' IO',5X,5(1PD11.3))
     WRITE(10,7030) (NITE(I),I=1,5)
7030 FORMAT(11X,' NITE',3X,5(1PD11.3))
     WRITE(10,7040) (NASTE(I),I=1,5)
7040 FORMAT(11X,' NASTE',2X,5(1PD11.3))
     WRITE(10,7050) (NSTEV(I),I=1,5)
7050 FORMAT(11X,' NSTEV',2X,5(1PD11.3))
```

```
      WRITE(10,7060) (SASI(I),I=1,5)
 )060 FORMAT(11X,' SASI',3X,5(1PD11.3))

         COMMENT OUT THESE WRITE'S FOR THE NON-ADAPTIVE
         CASE BECAUSE THE PARAMETERS DO NOT CHANGE

      WRITE(10,7070) (SACPEF(I),I=1,5)
 )70  FORMAT(11X,' SACPE F',5(1PD11.3))
      WRITE(10,7080) (SACPEG(I),I=1,5)
 )80  FORMAT(11X,' SACPE G',5(1PD11.3))
      WRITE(10,7090) (SCPEVF(I),I=1,5)
 )90  FORMAT(11X,' SCPEV F',5(1PD11.3))
      WRITE(10,7100) (SCPEVG(I),I=1,5)
 100  FORMAT(11X,' SCPEV G',5(1PD11.3))
      IF(NALG.GT.3) GOTO 7500
      WRITE(10,7110) (SAPPEA(I),I=1,5)
 110  FORMAT(11X,' SAPPE A',5(1PD11.3))
      WRITE(10,7120) (SAPPEB(I),I=1,5)
 120  FORMAT(11X,' SAPPE B',5(1PD11.3))
      WRITE(10,7130) (SPPEVA(I),I=1,5)
 130  FORMAT(11X,' SPPEV A',5(1PD11.3))
      WRITE(10,7140) (SPPEVB(I),I=1,5)
 140  FORMAT(11X,' SPPEV B',5(1PD11.3)/)
      GOTO 7800

 )00  CONTINUE
      WRITE(10,7750)
 '50  FORMAT(////)

 )00  CONTINUE
      WRITE(10,7000) (KD(I),I=7,11)
      WRITE(10,7010) (IDO(I),I=6,10)
      WRITE(10,7020) (IO(I),I=6,10)
      WRITE(10,7030) (NITE(I),I=6,10)
      WRITE(10,7040) (NASTE(I),I=6,10)
      WRITE(10,7050) (NSTEV(I),I=6,10)
      WRITE(10,7060) (SASI(I),I=6,10)

         COMMENT OUT THESE WRITE STATEMENTS FOR THE
         NON-ADAPTIVE CASE, SINCE F,G,A,B DO NOT VARY

      WRITE(10,7070) (SACPEF(I),I=6,10)
      WRITE(10,7080) (SACPEG(I),I=6,10)
      WRITE(10,7090) (SCPEVF(I),I=6,10)
      WRITE(10,7100) (SCPEVG(I),I=6,10)
      IF(NALG.GT.3) GOTO 8000
      WRITE(10,7110) (SAPPEA(I),I=6,10)
      WRITE(10,7120) (SAPPEB(I),I=6,10)
      WRITE(10,7130) (SPPEVA(I),I=6,10)
      WRITE(10,7140) (SPPEVB(I),I=6,10)
      RETURN
 )00  CONTINUE
      WRITE(10,7750)
      RETURN
      END
```

```
      SUBROUTINE INP(MODE,K,AMP,W,NSEED,U)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL RANNO
      IF(K.LT.0) GOTO 1000
      GOTO(100,200,300,400),MODE
  100 U = AMP
      RETURN
  200 CONTINUE
  300 U = AMP*DSIN(W*FLOAT(K))
      RETURN
  400 CALL RANDU(NSEED,M,RANNO)
      NSEED = M
      U = AMP*(RANNO-0.25)
      RETURN
 1000 U = 0.0D0
      RETURN
      END
```

```
      SUBROUTINE STAT(N,KOUNT)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 R(4),U(4),Y(4),S(4)
      REAL*8 IDO(10),IO(10),NITE(10),SASI(10),SAPPEA(10),
     * SAPPEB(10),SPPEVA(10),SPPEVB(10),SACPEF(10),SACPEG(10),
     * SCPEVF(10),SCPEVG(10),NASTE(10),NSTEV(10)
      REAL*8 AHAT(4),BHAT(4),FHAT(4),GHAT(4)
      INTEGER KD(12)
      COMMON /STATS/IDO,IO,NITE,SASI,SAPPEA,SAPPEB,
     * SPPEVA,SPPEVB,SACPEF,SACPEG,SCPEVF,SCPEVG,NASTE,NSTEV,
     * STE,SSPPA,SSPPB,SSCPF,SSCPG,M,KD
      COMMON /CONTRL/F,G,AHAT,BHAT,FHAT,GHAT
      COMMON /SIGNAL/R,U,Y,S
      DATA K/4/
      IF(DABS(S(K)).LT.1.0D-03) GOTO 100
      TE = (S(K)-Y(K))/S(K)
      GOTO 200
  100 TE = 0.0
      KOUNT = KOUNT+1
  200 NASTE(M) = NASTE(M)+TE**2
      STE = STE+TE
      SASI(M)   = SASI(M)+U(K)**2
      SAPPEA(M) = SAPPEA(M)+AHAT(K-1)
      SAPPEB(M) = SAPPEB(M)+BHAT(K-1)
      SACPEF(M) = SACPEF(M)+F
      SACPEG(M) = SACPEG(M)+G
      SSPPA = SSPPA+AHAT(K-1)**2
      SSPPB = SSPPB+BHAT(K-1)**2
      SSCPF = SSCPF+F**2
      SSCPG = SSCPG+G**2

  300 TIME = FLOAT(N)
      IF(N.LT.KD(M+1)) RETURN
      IDO(M) = S(K)
      IO(M)  = Y(K)
      RANGE  = FLOAT(KD(M+1)-KD(M))
      DIV = (RANGE-FLOAT(KOUNT))*(RANGE-FLOAT(KOUNT)-1)
      RNG = RANGE-FLOAT(KOUNT)
      NSTEV(M)  = 0.0
      IF(DIV.GT.0.0) NSTEV(M) = (RNG*NASTE(M)-STE**2)/DIV
      IF(NSTEV(M).LT.0.0D0) NSTEV(M) = 0.0D0
      NITE(M) = TE
      DIVSOR = RANGE*(RANGE-1)
      IF(DIVSOR.LE.0.0) GOTO 1000
      SPPEVA(M) = (RANGE*SSPPA-SAPPEA(M)**2)/DIVSOR
      IF(SPPEVA(M).LT.0.0D0) SPPEVA(M) = 0.0D0
      SPPEVB(M) = (RANGE*SSPPB-SAPPEB(M)**2)/DIVSOR
      IF(SPPEVB(M).LT.0.0D0) SPPEVB(M) = 0.0D0
      SCPEVF(M) = (RANGE*SSCPF-SACPEF(M)**2)/DIVSOR
      IF(SCPEVF(M).LT.0.0D0) SCPEVF(M) = 0.0D0
      SCPEVG(M) = (RANGE*SSCPG-SACPEG(M)**2)/DIVSOR
      IF(SCPEVG(M).LT.0.0D0) SCPEVG(M) = 0.0D0
      GOTO 2000
 1000 SPPEVA(M) = 0.0D0
      SPPEVB(M) = 0.0D0
```

```
      SCPEVF(M)  = 0.0D0
      SCPEVG(M)  = 0.0D0
000   CONTINUE
      SASI(M)  = SASI(M)/RANGE
      SAPPEA(M)  = SAPPEA(M)/RANGE
      SAPPEB(M)  = SAPPEB(M)/RANGE
      SACPEF(M)  = SACPEF(M)/RANGE
      SACPEG(M)  = SACPEG(M)/RANGE
      NASTE(M)   = NASTE(M)/RANGE
      KOUNT = 0
      STE = 0.0D0
      SSPPA = 0.0D0
      SSPPB = 0.0D0
      SSCPF = 0.0D0
      SSCPG = 0.0D0
      M = M+1
      RETURN
      END
```

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      GEI/SEI ALGORITHM

      EQUATIONS (4-4) - (4-8)

      SUBROUTINE ADAPT1(C,D,MU,RHO,H)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 MU,R(4),U(4),Y(4),S(4),AHAT(4),BHAT(4),E(4)
      REAL*8 FHAT(4),GHAT(4)
      COMMON /SIGNAL/R,U,Y,S
      COMMON /CONTRL/F,G,AHAT,BHAT,FHAT,GHAT
      DATA K/4/

      AHAT(K-2) = AHAT(K-1)
      AHAT(K-1) = AHAT(K)
      BHAT(K-2) = BHAT(K-1)
      BHAT(K-1) = BHAT(K)
      E(K-2) = E(K-1)
      E(K-1) = E(K)
      FHAT(K-2) = FHAT(K-1)
      FHAT(K-1) = FHAT(K)
      GHAT(K-2) = GHAT(K-1)
      GHAT(K-1) = GHAT(K)

      E(K-1) = Y(K-1)-AHAT(K-1)*Y(K-2)-BHAT(K-1)*U(K-2)
      DENOM  = H+MU*Y(K-2)**2+RHO*U(K-2)**2
      AHAT(K) = AHAT(K-1)+MU*Y(K-2)*E(K-1)/DENOM
      BHAT(K) = BHAT(K-1)+RHO*U(K-2)*E(K-1)/DENOM
      GHAT(K-1) = C/BHAT(K)
      FHAT(K-1) = (D-AHAT(K))/BHAT(K)
      F = FHAT(K-1)
      G = GHAT(K-1)

      RETURN
      END
```

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      GOI ALGORITHM

      EQUATIONS (4-12) - (4-19)

      SUBROUTINE ADAPT2(C,D,MU,RHO,H)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 MU,R(4),U(4),Y(4),S(4),YHAT(4),AHAT(4),BHAT(4)
      REAL*8 E(4),FHAT(4),GHAT(4)
      REAL*8 LAMBDA(4),GAMMA(4)
      COMMON /SIGNAL/R,U,Y,S
      COMMON /CONTRL/F,G,AHAT,BHAT,FHAT,GHAT
      COMMON /MISC1/GAMMA,LAMBDA,YHAT
      DATA K/4/

      LAMBDA(K-2)  =  LAMBDA(K-1)
      LAMBDA(K-1)  =  LAMBDA(K)
      GAMMA(K-2)   =  GAMMA(K-1)
      GAMMA(K-1)   =  GAMMA(K)
      AHAT(K-2)  =  AHAT(K-1)
      AHAT(K-1)  =  AHAT(K)
      BHAT(K-2)  =  BHAT(K-1)
      BHAT(K-1)  =  BHAT(K)
      YHAT(K-2)  =  YHAT(K-1)
      YHAT(K-1)  =  YHAT(K)
      E(K-2)  =  E(K-1)
      E(K-1)  =  E(K)
      FHAT(K-2)  =  FHAT(K-1)
      FHAT(K-1)  =  FHAT(K)
      GHAT(K-2)  =  GHAT(K-1)
      GHAT(K-1)  =  GHAT(K)

      YHAT(K-1)  =  AHAT(K-1)*YHAT(K-2)+BHAT(K-1)*U(K-2)
      LAMBDA(K-1)  =  YHAT(K-2)+AHAT(K-1)*LAMBDA(K-2)
      GAMMA(K-1)   =  U(K-2)+AHAT(K-1)*GAMMA(K-2)
      E(K-1)  =  Y(K-1)-YHAT(K-1)
      DENOM   =  H+MU*LAMBDA(K-1)**2+RHO*GAMMA(K-1)**2
      AHAT(K)  =  AHAT(K-1)+MU*LAMBDA(K-1)*E(K-1)/DENOM
      BHAT(K)  =  BHAT(K-1)+RHO*GAMMA(K-1)*E(K-1)/DENOM
      GHAT(K-1)  =  C/BHAT(K)
      FHAT(K-1)  =  (D-AHAT(K))/BHAT(K)
      F  =  FHAT(K-1)
      G  =  GHAT(K-1)

      RETURN
      END
```

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      SOI ALGORITHM

      EQUATIONS (4-23) - (4-29)

      SUBROUTINE ADAPT3(C,D,MU,RHO,H,Q)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 MU,R(4),U(4),Y(4),S(4),YHAT(4),AHAT(4),BHAT(4)
      REAL*8 V(4),Z(4),FHAT(4),GHAT(4)
      COMMON /SIGNAL/R,U,Y,S
      COMMON /CONTRL/F,G,AHAT,BHAT,FHAT,GHAT
      COMMON /MISC2/YHAT,Z,V
      DATA K/4/

      Z(K-2)  = Z(K-1)
      Z(K-1)  = Z(K)
      AHAT(K-2)  = AHAT(K-1)
      AHAT(K-1)  = AHAT(K)
      BHAT(K-2)  = BHAT(K-1)
      BHAT(K-1)  = BHAT(K)
      YHAT(K-2)  = YHAT(K-1)
      YHAT(K-1)  = YHAT(K)
      V(K-2)  = V(K-1)
      V(K-1)  = V(K)
      FHAT(K-2)  = FHAT(K-1)
      FHAT(K-1)  = FHAT(K)
      GHAT(K-2)  = GHAT(K-1)
      GHAT(K-1)  = GHAT(K)

      YHAT(K-1)  = AHAT(K-1)*Z(K-2)+BHAT(K-1)*U(K-2)
      DENOM   = H+MU*Z(K-2)**2+RHO*U(K-2)**2
      V(K-1)  = (Y(K-1)-YHAT(K-1)+Q*(Y(K-2)-Z(K-2)))/DENOM
      AHAT(K)  = AHAT(K-1)+MU*Z(K-2)*V(K-1)
      BHAT(K)  = BHAT(K-1)+RHO*U(K-2)*V(K-1)
      Z(K-1)  = AHAT(K)*Z(K-2)+BHAT(K)*U(K-2)
      GHAT(K-1)  = C/BHAT(K)
      FHAT(K-1)  = (D-AHAT(K))/BHAT(K)
      F = FHAT(K-1)
      G = GHAT(K-1)

      RETURN
      END
```

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      GED/SED ALGORITHM

      EQUATIONS (4-34) - (4-36)

      SUBROUTINE ADAPT4(C,D,MU,RHO,H)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 MU,R(4),U(4),Y(4),S(4),V(4),FHAT(4),GHAT(4)
      REAL*8 AHAT(4),BHAT(4)
      COMMON /SIGNAL/R,U,Y,S
      COMMON /CONTRL/F,G,AHAT,BHAT,FHAT,GHAT
      DATA K/4/

      GHAT(K-2) = GHAT(K-1)
      GHAT(K-1) = GHAT(K)
      FHAT(K-2) = FHAT(K-1)
      FHAT(K-1) = FHAT(K)
      V(K-2) = V(K-1)
      V(K-1) = V(K)

      V(K-1) = C*R(K-2)+D*Y(K-2)-Y(K-1)
      DENOM = H*(1.0+MU*Y(K-2)**2+RHO*U(K-2)**2)
      GHAT(K-1) = GHAT(K-2)+RHO*R(K-2)*V(K-1)/DENOM
      FHAT(K-1) = FHAT(K-2)+MU*Y(K-2)*V(K-1)/DENOM
      G = GHAT(K-1)
      F = FHAT(K-1)

      RETURN
      END
```

```
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccc

      SOD ALGORITHM

      EQUATIONS (4-40) - (4-44)

      SUBROUTINE ADAPT5(C,D,MU,RHO,H,Q)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 MU,R(4),U(4),Y(4),S(4),V(4),FHAT(4),GHAT(4)
      REAL*8 GAMMA(4),BETA(4),AHAT(4),BHAT(4)
      COMMON /SIGNAL/R,U,Y,S
      COMMON /CONTRL/F,G,AHAT,BHAT,FHAT,GHAT
      COMMON /MISC3/BETA,GAMMA,V
      DATA K/4/

      GHAT(K-2)  = GHAT(K-1)
      GHAT(K-1)  = GHAT(K)
      FHAT(K-2)  = FHAT(K-1)
      FHAT(K-1)  = FHAT(K)
      BETA(K-2)  = BETA(K-1)
      BETA(K-1)  = BETA(K)
      GAMMA(K-2)  = GAMMA(K-1)
      GAMMA(K-1)  = GAMMA(K)
      V(K-2)  = V(K-1)
      V(K-1)  = V(K)

      BETA(K-1)  = RHO*R(K-2)**2+MU*Y(K-2)**2
      GAMMA(K-1)  =  (D+Q)*H*BETA(K-2)*V(K-2)+D*GAMMA(K-2)
      DENOM = 1.0+H*BETA(K-1)
      V(K-1)  =  (S(K-1)-Y(K-1)+Q*(S(K-2)-Y(K-2))-GAMMA(K-1))/DENOM
      GHAT(K-1)  = GHAT(K-2)+RHO*R(K-2)*V(K-1)
      FHAT(K-1)  = FHAT(K-2)+MU*Y(K-2)*V(K-1)
      G = GHAT(K-1)
      F = FHAT(K-1)

      RETURN
      END
```

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C       THIS PROGRAM PROVIDES THE INITIAL VALUES NECESSARY
C       TO SIMULATE EXAMPLES 5 AND 6. THE SIMULATION IS
C       PROVIDED BY SUBROUTINE SIMUL2.
C
C
        BLOCK DATA
        IMPLICIT REAL*8 (A-H,O-Z)
        REAL*8 IDD(10),ID(10),NITE(10),SASI(10),
     *    SAPEA1(10),SAPEA2(10),SAPEB1(10),SAPEB2(10),
     *    SPEVA1(10),SPEVA2(10),SPEVB1(10),SPEVB2(10),
     *    SACEE1(10),SACEE2(10),SACEN1(10),SACEN2(10),
     *    SCEVE1(10),SCEVE2(10),SCEVN1(10),SCEVN2(10),
     *    NASTE(10),NSTEV(10)
        INTEGER KD(12)
        COMMON /STATS/IDD,ID,NITE,SASI,SAPEA1,SAPEA2,SAPEB1,SAPEB2,
     *    SPEVA1,SPEVA2,SPEVB1,SPEVB2,SACEE1,SACEE2,SACEN1,SACEN2,
     *    SCEVE1,SCEVE2,SCEVN1,SCEVN2,
     *    NASTE,NSTEV,STE,
     *    SSPA1,SSPA2,SSPB1,SSPB2,SSCE1,SSCE2,SSCN1,SSCN2,M,KD
        DATA KD/0,1,2,5,10,20,50,100,200,500,1000,2000/
        END
C
        IMPLICIT REAL*8 (A-H,O-Z)
        EXTERNAL UVFL
        REAL*8 A1(5),A2(5),B1(5),B2(5),AMP(4)
        REAL*8 ALPHA(2,2),BETA(2,2)
        REAL*8 A(5),B(5)
        REAL*8 OMEGA(2),ZETA(2),LAMBDA(2),GAMMA(2)
        REAL*8 EXPOWT,COSWT,T
        REAL*8 ESTMUL(3)
        REAL YMIN(4),YMAX(4)
        INTEGER PCIL,PTEMP,PXCHG,ESTSTR(3),SCSTR(3)
        REAL*8 IDD(10),ID(10),NITE(10),SASI(10),
     *    SAPEA1(10),SAPEA2(10),SAPEB1(10),SAPEB2(10),
     *    SPEVA1(10),SPEVA2(10),SPEVB1(10),SPEVB2(10),
     *    SACEE1(10),SACEE2(10),SACEN1(10),SACEN2(10),
     *    SCEVE1(10),SCEVE2(10),SCEVN1(10),SCEVN2(10),
     *    NASTE(10),NSTEV(10)
        REAL*8 ETA1(5),ETA2(5),NO1(5),NO2(5)
        INTEGER KD(12)
C
        COMMON /STATS/IDD,ID,NITE,SASI,SAPEA1,SAPEA2,SAPEB1,SAPEB2,
     *    SPEVA1,SPEVA2,SPEVB1,SPEVB2,SACEE1,SACEE2,SACEN1,SACEN2,
     *    SCEVE1,SCEVE2,SCEVN1,SCEVN2,
     *    NASTE,NSTEV,STE,
     *    SSPA1,SSPA2,SSPB1,SSPB2,SSCE1,SSCE2,SSCN1,SSCN2,M,KD
        COMMON /COEFF/A1,A2,B1,B2,ETA1,ETA2,NO1,NO2,NACT
        COMMON /LCN/
C
        DATA YMIN/-10.0,-10.0,-10.0,-10.0/
        DATA YMAX/10.0,10.0,10.0,10.0/
        DATA MAX,SEED/1000,13579.D0/
C       SET PARAMETERS FOR EXAMPLES 5 AND 6
        DATA SIGMA/1.0/
        DATA OMEGA/0.5,5.0/
        DATA ZETA/0.2,0.02/
        DATA GAMMA/1.607,-0.741/
        DATA LAMBDA/1.0,1.0/
        DATA DELTA/0.5/
        DATA T/0.1/
        DATA AMP/2.0D0,2.0D0,2.0D0,2.0D0/
        DATA ESTMUL/0.8D0,1.0D0,1.2D0/
        DATA PCIL/'1'/
C       DATA PTEMP/'0'/
        DATA ESTSTR/'-20%',' 0  ','+20%'/
        DATA SCSTR/'SC- ','SC0 ','SC+ '/
C
        LP = 6
        CALL ERRSET(207,1000,-1,1,UVFL,0)
        CALL ERRSET(208,1000,-1,1,1,0)
C
C       CALCULATE COEFFICIENTS FOR THE Z-PLANE EQUIVILENT OF
C       THE 4TH ORDER CONTINUOUS PLANT (EQNS. (5-18)-(5-22))
        DO 5 I=1,2
        EXPOWT = DEXP(-ZETA(I)*OMEGA(I)*T)
        COSWT  = DCOS(OMEGA(I)*T*DSQRT(1.0-ZETA(I)**2))
        ALPHA(I,1) = 2.0*EXPOWT*COSWT
        ALPHA(I,2) = -(EXPOWT**2)
        BETA(I,1)  = (LAMBDA(I)/OMEGA(I)**2)*(1.0-EXPOWT*
     *               (COSWT+(ZETA(I)/DSQRT(1.0-ZETA(I)**2))*
     *               DSIN(OMEGA(I)*T*DSQRT(1.0-ZETA(I)**2))))
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      BETA(I,2)  = (LAMBDA(I)/OMEGA(I)**2)*(LEXPCWT*
     *            (LEXPCWT-COSWT+(ZETA(I)/DSQRT(1.0-ZETA(I)**2))*
     *             DSIN(OMEGA(I)*T*DSQRT(1.0-ZETA(I)**2))))
    5 CONTINUE
C
C     CALCULATE ACTUATOR COEFFICIENTS
      ESIG1 = DEXP(-SIGMA*T)
      ESIG2 = 1.-DEXP(-SIGMA*T)
C
C     DEFINE NEW DIFFERENCE EQUATION PARAMETERS FOR THE 4TH ORDER PLANT
C     FROM EQN. (4-94).
      A(1) = ALPHA(1,1)+ALPHA(2,1)
      A(2) = ALPHA(1,2)-ALPHA(1,1)*ALPHA(2,1)+ALPHA(2,2)
      A(3) = -ALPHA(1,2)*ALPHA(2,1)-ALPHA(1,1)*ALPHA(2,2)
      A(4) = -ALPHA(1,2)*ALPHA(2,2)
      B(1) = BETA(1,1)+BETA(2,1)
      B(2) = BETA(1,2)+BETA(2,2)-ALPHA(2,1)*BETA(1,1)
     *       -ALPHA(1,1)*BETA(2,1)
      B(3) = -ALPHA(2,2)*BETA(1,1)-ALPHA(2,1)*BETA(1,2)
     *       -ALPHA(1,2)*BETA(2,1)-ALPHA(1,1)*BETA(2,2)
      B(4) = -ALPHA(2,2)*BETA(1,2)-ALPHA(1,2)*BETA(2,2)
C
      WRITE(6,10)
   10 FORMAT(' PRINTOUT INTERVAL = ')
      READ(5,*)INTV
C
      WRITE(6,30)
   30 FORMAT(' STARTING AND ENDING ALGORITHMS = ')
      READ(5,*)NALGST,NALGED
      WRITE(6,40)
   40 FORMAT(' SMOOTHING COEFFICIENT FOR SUI ALGORITHM (#3) = ')
      READ(5,*)NSC
      NEX = 5
C
      WRITE(6,35)
   35 FORMAT(' ADAPTIVE STEP SIZE = ')
      READ(5,37) STPSIZ
   37 FORMAT(F4.2)
C
C     ACTUATOR CONFIGURATION LOOP (1=NO ACTUATOR,
C     2=ACTUATOR CONFIGURATION (A), 3=ACTUATOR
C     CONFIG. (B), 4=ACTUATOR CONFIG. (C) ).
      DO 100 NACT=1,4
C
C     ADAPTIVE ALGORITHMS LOOP
   42 DO 100 NALG=NALGST,NALGED
C
C     INITIAL PLANT PARAMETER ESTIMATES LOOP
      DO 100 NEST=1,3
C
C     INPUTS LOOP
      DO 100 INP=1,4
C
      IF(NACT.GT.1) NEX = 6
C
C     SET INITIAL PARAMETER ESTIMATES (ALPHA(1,1) BECOMES
C     A1(1)=ALPHA(1)-HAT ETC., IN EQNS. (4-69)-(4-92)).
C
      DO 45 I=1,5
      A1(I) = ALPHA(I,1)*ESTMUL(NEST)
      A2(I) = ALPHA(I,2)*ESTMUL(NEST)
      B1(I) = BETA(I,1)*ESTMUL(NEST)
      B2(I) = BETA(I,2)*ESTMUL(NEST)
   45 CONTINUE
C
C     CALCULATE INITIAL CONTROLLER PARAMETERS
      CALL CNTSET(GAMMA)
C
      WRITE(10,50)PCTL,NEX,INP,NALG,ESTSTR(NEST),SCSTR(NSC),
     *  NACT,STPSIZ
   50 FORMAT(A1,2X,'EXAMPLE',I2,'  INPUT',I2,' ALGORITHM',I2,
     *  ' ES1 = ',A4,' SS2 =',A4,' NACT=',I2,' ST =',F4.2/)
      WRITE(6,50)PCTL,NEX,INP,NALG,ESTSTR(NEST),SCSTR(NSC),NACT
     * ,STPSIZ
C     PXCHG = PCTL
C     PCTL  = PTEMP
C     ENTER SIMULATION SUBROUTINE
C     PTEMP = PXCHG
      CALL SIMUL2(ALPHA,BETA,A,B,GAMMA,DELTA,AMP(INP),
     *  DSEED,INP,NALG,NEX,NSC,YMIN(INP),YMAX(INP),
     *  INTV,MAX,ESIG1,ESIG2,STPSIZ)
  100 CONTINUE
      ENDFILE 10
```

```
      STOP
      END
C
      SUBROUTINE OVFL
      LOGICAL OVFLOW
      COMMON /BOMB/OVFLOW
      WRITE(6,10)
   10 FORMAT(' *** OVERFLOW ***'/)
      OVFLOW = .TRUE.
      RETURN
      END

      SUBROUTINE SIMUL2(ALPHA,BETA,A,B,GAMMA,DELTA,AMP,DSEED,
     * MODE,NALG,NEX,NSC,YMIN,YMAX,INIV,MAX,ESIG1,ESIG2,STPSIZ)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 ALPHA(2,2),BETA(2,2)
      REAL*8 A(5),B(5)
      REAL*8 U(5),Y(5),R(5),S(5),UC(5)
      REAL*8 MU(2),RHO(2),GAMMA(2),C(2)
      REAL SIG(5),TIME
      INTEGER SYMBOL(4)
      REAL*8 T1(5),T2(5),T3(5),T4(5),T5(5),T6(5),T7(5),
     * T8(5)
      REAL*8 SC(3,2)
      REAL*8 IDU(10),IU(10),NITE(10),SASI(10),
     * SAPEA1(10),SAPEA2(10),SAPEE1(10),SAPEE2(10),
     * SPEVA1(10),SPEVA2(10),SPEVE1(10),SPEVE2(10),
     * SACEE1(10),SACEE2(10),SACEN1(10),SACEN2(10),
     * SCEVE1(10),SCEVE2(10),SCEVN1(10),SCEVN2(10),
     * NASTE(10),NSTEV(10)
      REAL*8 A1(5),A2(5),B1(5),B2(5),ETA1(5),ETA2(5),NU1(5),NU2(5)
      INTEGER KD(12)
      LOGICAL OVFLOW
C
      COMMON /STATS/IDU,IU,NITE,SASI,SAPEA1,SAPEA2,SAPEE1,SAPEE2,
     * SPEVA1,SPEVA2,SPEVE1,SPEVE2,SACEE1,SACEE2,SACEN1,SACEN2,
     * SCEVE1,SCEVE2,SCEVN1,SCEVN2,
     * NASTE,NSTEV,STE,
     * SSPA1,SSPA2,SSPB1,SSPB2,SSCE1,SSCE2,SSCN1,SSCN2,M,KD
C
      COMMON /LUN/LP
      COMMON /BOMB/OVFLOW
      COMMON /SIGNAL/R,U,Y,S,UC
      COMMON /CONTRL/A1,A2,B1,B2,ETA1,ETA2,NU1,NU2,NACT
      COMMON /MISC1/T1,T2,T3,T4,T5
      COMMON /MISC2/T6,T7,T8
C
      DATA K/5/
      DATA SYMBOL/'U','R','S','Y'/
      DATA SC(1,1),SC(1,2)/-0.9,0.0/
      DATA SC(2,1),SC(2,2)/-1.840,0.9515/
      DATA SC(3,1),SC(3,2)/-0.90,0.0/
C
      OVFLOW = .FALSE.
C
      DO 100 I=1,5
      U(I) = 0.0D0
      UC(I) = 0.0D0
      Y(I) = 0.0D0
      R(I) = 0.0D0
      S(I) = 0.0D0
      T1(I) = 0.0D0
      T2(I) = 0.0D0
      T3(I) = 0.0D0
      T4(I) = 0.0D0
      T5(I) = 0.0D0
      T6(I) = 0.0D0
      T7(I) = 0.0D0
      T8(I) = 0.0D0
  100 CONTINUE
      DSD = DSEED
      M = 1
      DO 110 I=1,10
      IDU(I) = 0.0D0
      IU(I)  = 0.0D0
      SASI(I)  = 0.0D0
      NASTE(I) = 0.0D0
      NSTEV(I) = 0.0D0
      NITE(I)  = 0.0D0
```

```
      SAPEA1(I) = 0.0D0
      SAPEA2(I) = 0.0D0
      SAPEB1(I) = 0.0D0
      SAPEB2(I) = 0.0D0
      SACEE1(I) = 0.0D0
      SACEE2(I) = 0.0D0
      SACEN1(I) = 0.0D0
      SACEN2(I) = 0.0D0
      SPEVA1(I) = 0.0D0
      SPEVA2(I) = 0.0D0
      SPEVB1(I) = 0.0D0
      SPEVB2(I) = 0.0D0
      SCEVE1(I) = 0.0D0
      SCEVE2(I) = 0.0D0
      SCEVN1(I) = 0.0D0
      SCEVN2(I) = 0.0D0
  110 CONTINUE
      KOUNT = 0
      STE   = 0.0D0
      SSPA1 = 0.0D0
      SSPA2 = 0.0D0
      SSPB1 = 0.0D0
      SSPB2 = 0.0D0
      SSCE1 = 0.0D0
      SSCE2 = 0.0D0
      SSCN1 = 0.0D0
      SSCN2 = 0.0D0
      IPRT = 0
      INIT = 1
C
      MU(1)  = STPS12
      MU(2)  = STPS12
      RHO(1) = STPS12
      RHO(2) = STPS12
C
      H = 1.0D0
C
      Q(1) = SC(NSC,1)
      Q(2) = SC(NSC,2)
C
C     ENTER MAIN SIMULATION LOOP
C
      DO 5000 N=1,MAX
C
      IF(OVFLOW) GOTO 6000
C
      Y(K) = 0.0D0
C
C     CALCULATE NEXT PLANT OUTPUT
      DO 150 I=1,4
  150 Y(K) = Y(K)+A(I)*Y(K-I)+B(I)*U(K-I)
C
C     CALCULATE DESIRED OUTPUT
      S(K) = DELTA*(BETA(1,1)*R(K-1)+BETA(1,2)*R(K-2))
     *       +GAMMA(1)*S(K-1)+GAMMA(2)*S(K-2)
C
C     COMPUTE STATISTICS AND PRINT VALUES
C
      TIME = FLOAT(N)
C
      CALL STAT2(N,KOUNT)
C
      IF((N/INTV)*INTV.NE.N) GOTO 1000
      SIG(1) = SNGL(UC(K-1))
      SIG(2) = SNGL(R(K-1))
      SIG(3) = SNGL(S(K))
      SIG(4) = SNGL(Y(K))
      CALL TIPLOT(TIME,SIG,4,SYMBOL,YMIN,YMAX,IPRT,INIT)
C
C     SHIFT VARIABLES TO ALLOW CALCULATION OF NEW VALUES
 1000 R(K-4) = R(K-3)
      R(K-3) = R(K-2)
      R(K-2) = R(K-1)
      U(K-4) = U(K-3)
      U(K-3) = U(K-2)
      U(K-2) = U(K-1)
      UC(K-4) = UC(K-3)
      UC(K-3) = UC(K-2)
      UC(K-2) = UC(K-1)
```

```
      Y(K-4) = Y(K-3)
      Y(K-3) = Y(K-2)
      Y(K-2) = Y(K-1)
      Y(K-1) = Y(K)
      S(K-4) = S(K-3)
      S(K-3) = S(K-2)
      S(K-2) = S(K-1)
      S(K-1) = S(K)
C     UPDATE PLANT PARAMETER ESTIMATES
C
      GOTO (210,220,230),NALG
  210 CALL GE1(GAMMA,MU,RHO,F )
      GOTO 300
  220 CALL GU1(GAMMA,MU,RHO,H)
      GOTO 300
  230 CALL SU1(GAMMA,MU,RHO,H,C)
      GOTO 300
  300 CONTINUE
C
C
C     FIND NEW INPUT R(K-1)
C
      CALL INP(MODE,N,AMP,DSD,R(K-1))
C
C     FIND NEW CONTROLLER OUTPUT UC(K-1) AND ACTUATOR OUTPUT U(K-1)
C
      IF(NACT.EQ.3) GOTO 410
C
C     CALCULATIONS FOR ACTUATOR CONFIG.1, 2 AND 4
      UC(K-1) = DELTA*R(K-1)+ETA1(K-1)*UC(K-2)+ETA2(K-1)*UC(K-3)
     *       +NU1(K-1)*Y(K-2)+NU2(K-1)*Y(K-3)
C
      IF(NACT.GT.1) GOTO 420
      GOTO 415
C
C     CALCULATION FOR ACTUATOR CONFIG. 3:
  410 UC(K-1) = DELTA*R(K-1)+ETA1(K-1)*U(K-2)+ETA2(K-1)*U(K-3)
     *       +NU1(K-1)*Y(K-2)+NU2(K-1)*Y(K-3)
  415 U(K-1) = ESIG1*U(K-2)+ESIG2*UC(K-2)
      GOTO 5000
  420 U(K-1) = UC(K-1)
 5000 CONTINUE
      GOTO 6500
 6000 CONTINUE
      IDD(M)    = 0.0D0
      ID(M)     = 0.0D0
      SASI(M)   = 0.0D0
      NASIE(M)  = 0.0D0
      NSTEV(M)  = 0.0D0
      NITE(M)   = 0.0D0
      SAPEA1(M) = 0.0D0
      SAPEA2(M) = 0.0D0
      SAPEB1(M) = 0.0D0
      SAPEB2(M) = 0.0D0
      SACEE1(M) = 0.0D0
      SACEE2(M) = 0.0D0
      SACEN1(M) = 0.0D0
      SACEN2(M) = 0.0D0
      SPEVA1(M) = 0.0D0
      SPEVA2(M) = 0.0D0
      SPEVB1(M) = 0.0D0
      SPEVB2(M) = 0.0D0
      SCEVE1(M) = 0.0D0
      SCEVE2(M) = 0.0D0
      SCEVN1(M) = 0.0D0
      SCEVN2(M) = 0.0D0
 6500 CONTINUE
      WRITE(10,7000)(KD(I),I=2,6)
 7000 FORMAT(11X,' K',7X,5(2X,I5,4X)/)
      WRITE(10,7010)(IDD(I),I=1,5)
 7010 FORMAT(11X,' IDD',5X,5(1PD11.3))
      WRITE(10,7020)(ID(I),I=1,5)
 7020 FORMAT(11X,' ID',6X,5(1PD11.3))
      WRITE(10,7030)(NITE(I),I=1,5)
 7030 FORMAT(11X,' NITE',4X,5(1PD11.3))
      WRITE(10,7040)(NASIE(I),I=1,5)
 7040 FORMAT(11X,' NASIE',3X,5(1PD11.3))
      WRITE(10,7050)(NSTEV(I),I=1,5)
 7050 FORMAT(11X,' NSTEV',3X,5(1PD11.3))
      WRITE(10,7060)(SASI(I),I=1,5)
 7060 FORMAT(11X,' SASI',4X,5(1PD11.3))
      WRITE(10,7070)(SACEE1(I),I=1,5)
```

```
 7070 FORMAT(11X,' SACPE E1',5(1PD11.3))
      WRITE(10,7075)(SACEE2(I),I=1,5)
 7075 FORMAT(11X,' SACPE E2',5(1PD11.3))
      WRITE(10,7080)(SACEN1(I),I=1,5)
 7080 FORMAT(11X,' SACPE N1',5(1PD11.3))
      WRITE(10,7085)(SACEN2(I),I=1,5)
 7085 FORMAT(11X,' SACPE N2',5(1PD11.3))
      WRITE(10,7090)(SCEVE1(I),I=1,5)
 7090 FORMAT(11X,' SCPEV E1',5(1PD11.3))
      WRITE(10,7095)(SCEVE2(I),I=1,5)
 7095 FORMAT(11X,' SCPEV E2',5(1PD11.3))
      WRITE(10,7100)(SCEVN1(I),I=1,5)
 7100 FORMAT(11X,' SCPEV N1',5(1PD11.3))
      WRITE(10,7105)(SCEVN2(I),I=1,5)
 7105 FORMAT(11X,' SCPEV N2',5(1PD11.3))
      WRITE(10,7110)(SAPEA1(I),I=1,5)
 7110 FORMAT(11X,' SAPPE A1',5(1PD11.3))
      WRITE(10,7115)(SAPEA2(I),I=1,5)
 7115 FORMAT(11X,' SAPPE A2',5(1PD11.3))
      WRITE(10,7120)(SAPEB1(I),I=1,5)
 7120 FORMAT(11X,' SAPPE B1',5(1PD11.3))
      WRITE(10,7125)(SAPEB2(I),I=1,5)
 7125 FORMAT(11X,' SAPPE B2',5(1PD11.3))
      WRITE(10,7130)(SPEVA1(I),I=1,5)
 7130 FORMAT(11X,' SPPEV A1',5(1PD11.3))
      WRITE(10,7135)(SPEVA2(I),I=1,5)
 7135 FORMAT(11X,' SPPEV A2',5(1PD11.3))
      WRITE(10,7140)(SPEVB1(I),I=1,5)
 7140 FORMAT(11X,' SPPEV B1',5(1PD11.3))
      WRITE(10,7145)(SPEVB2(I),I=1,5)
 7145 FORMAT(11X,' SPPEV B2',5(1PD11.3)/)
      GOTO 7600
 7500 CONTINUE
      WRITE(10,7750)
 7750 FORMAT(////)
 7600 CONTINUE
      WRITE(10,7000)(RO(I),I=7,11)
      WRITE(10,7010)(IDD(I),I=6,10)
      WRITE(10,7020)(IE(I),I=6,10)
      WRITE(10,7030)(NITE(I),I=6,10)
      WRITE(10,7040)(NASTE(I),I=6,10)
      WRITE(10,7050)(NSTEV(I),I=6,10)
      WRITE(10,7060)(SAST(I),I=6,10)
      WRITE(10,7070)(SACEE1(I),I=6,10)
      WRITE(10,7075)(SACEE2(I),I=6,10)
      WRITE(10,7080)(SACEN1(I),I=6,10)
      WRITE(10,7085)(SACEN2(I),I=6,10)
      WRITE(10,7090)(SCEVE1(I),I=6,10)
      WRITE(10,7095)(SCEVE2(I),I=6,10)
      WRITE(10,7100)(SCEVN1(I),I=6,10)
      WRITE(10,7105)(SCEVN2(I),I=6,10)
      WRITE(10,7110)(SAPEA1(I),I=6,10)
      WRITE(10,7115)(SAPEA2(I),I=6,10)
      WRITE(10,7120)(SAPEB1(I),I=6,10)
      WRITE(10,7125)(SAPEB2(I),I=6,10)
      WRITE(10,7130)(SPEVA1(I),I=6,10)
      WRITE(10,7135)(SPEVA2(I),I=6,10)
      WRITE(10,7140)(SPEVB1(I),I=6,10)
      WRITE(10,7145)(SPEVB2(I),I=6,10)
      RETURN
 8000 CONTINUE
      WRITE(10,7750)
      RETURN
      END
C
      SUBROUTINE INP(MODE,K,AMP,DSEED,U)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL RANNO(1)
      M = 1
      IF(K.LT.0) GOTO 1000
      GOTO(100,200,300,400),MODE
  100 U = AMP
      RETURN
  200 U = AMP*DSIN(DBLE(0.20*FLOAT(K)))
      RETURN
  300 U = AMP*DSIN(DBLE(0.5*FLOAT(K)))
      RETURN
  400 CALL GGUBS(DSEED,M,RANNO)
      U = AMP*(RANNO(1)-0.25)
      RETURN
 1000 U = 0.0D0
      RETURN
      END
```

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C       GEI/SEI ALGORITHM
C
C       EQUATIONS (4-50) - (4-55)
C
        SUBROUTINE GEI(GAMMA,MU,RHO,H)
        IMPLICIT REAL*8 (A-H,O-Z)
        REAL*8 R(5),U(5),Y(5),S(5),A1(5),A2(5),B1(5),B2(5),UC(5)
        REAL*8 MU(2),RHO(2),GAMMA(2)
        REAL*8 E(5),V(5)
        REAL*8 ETA1(5),ETA2(5),NU1(5),NU2(5)
        COMMON /SIGNAL/R,U,Y,S,UC
        COMMON /CONTRL/A1,A2,B1,B2,ETA1,ETA2,NU1,NU2,NACT
        DATA K/5/
C
C       SHIFT VARIABLES TO ALLOW CALCULATION OF NEW VALUES
        A1(K-2) = A1(K-1)
        A1(K-1) = A1(K)
        A2(K-2) = A2(K-1)
        A2(K-1) = A2(K)
        B1(K-2) = B1(K-1)
        B1(K-1) = B1(K)
        B2(K-2) = B2(K-1)
        B2(K-1) = B2(K)
        E(K-2) = E(K-1)
        E(K-1) = E(K)
        V(K-2) = V(K-1)
        V(K-1) = V(K)
C
        IF(NACT.GT.2) GOTO 560
C
C       CALCULATIONS FOR ACTUATOR CONFIG. 1 AND 2:
        E(K-1) = Y(K-1)-A1(K-1)*Y(K-2)-A2(K-2)*Y(K-3)
     *      -B1(K-1)*UC(K-2)-B2(K-2)*UC(K-3)
        DENOM  = H+MU(1)*Y(K-2)**2+MU(2)*Y(K-3)**2
     *      +RHO(1)*UC(K-2)**2+RHO(2)*UC(K-3)**2
        GOTO 570
C
C       CALCULATIONS FOR ACTUATOR CONFIG. 3 AND 4
  560   E(K-1) = Y(K-1)-A1(K-1)*Y(K-2)-A2(K-2)*Y(K-3)
     *      -B1(K-1)*U(K-2)-B2(K-2)*U(K-3)
        DENOM  = H+MU(1)*Y(K-2)**2+MU(2)*Y(K-3)**2
     *      +RHO(1)*U(K-2)**2+RHO(2)*U(K-3)**2
  570   V(K-1) = E(K-1)/DENOM
C
C       CALCULATE NEW PLANT PARAMETER ESTIMATES
        A1(K) = A1(K-1)+MU(1)*V(K-1)*Y(K-2)
        A2(K) = A2(K-1)+MU(2)*V(K-1)*Y(K-3)
        B1(K) = B1(K-1)+RHO(1)*V(K-1)*U(K-2)
        B2(K) = B2(K-1)+RHO(2)*V(K-1)*U(K-3)
C
C       UPDATE CONTROLLER PARAMETER ESTIMATES
        CALL CNTSET(GAMMA)
        RETURN
        END


CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C       GDI ALGORITHM
C
C       EQUATIONS (4-61) - (4-70)
C
        SUBROUTINE GDI(GAMMA,MU,RHO,H)
        IMPLICIT REAL*8 (A-H,O-Z)
        REAL*8 R(5),U(5),Y(5),UC(5),S(5),YHAT(5)
        REAL*8 A1(5),A2(5),B1(5),B2(5)
        REAL*8 MU(2),RHO(2),GAMMA(2)
        REAL*8 E(5),V(5),ETA1(5),ETA2(5),NU1(5),NU2(5)
        REAL*8 LAM1(5),LAM2(5),GAM1(5),GAM2(5)
        COMMON /SIGNAL/R,U,Y,S,UC
        COMMON /CONTRL/A1,A2,B1,B2,ETA1,ETA2,NU1,NU2,NACT
        COMMON /MISC1/GAM1,GAM2,LAM1,LAM2,YHAT
        DATA K/5/
C
C       SHIFT VARIABLES TO ALLOW NEW VALUES TO BE CALCULATED
        LAM1(K-3) = LAM1(K-2)
        LAM1(K-2) = LAM1(K-1)
```

```
      LAM1(K-1) = LAM1(K)
      LAM2(K-3) = LAM2(K-2)
      LAM2(K-2) = LAM2(K-1)
      LAM2(K-1) = LAM2(K)
      GAM1(K-3) = GAM1(K-2)
      GAM1(K-2) = GAM1(K-1)
      GAM1(K-1) = GAM1(K)
      GAM2(K-3) = GAM2(K-2)
      GAM2(K-2) = GAM2(K-1)
      GAM2(K-1) = GAM2(K)
      A1(K-2) = A1(K-1)
      A1(K-1) = A1(K)
      A2(K-2) = A2(K-1)
      A2(K-1) = A2(K)
      B1(K-2) = B1(K-1)
      B1(K-1) = B1(K)
      B2(K-2) = B2(K-1)
      B2(K-1) = B2(K)
      YHAT(K-3) = YHAT(K-2)
      YHAT(K-2) = YHAT(K-1)
      YHAT(K-1) = YHAT(K)
      E(K-2) = E(K-1)
      E(K-1) = E(K)
C
      IF(NACT.GT.2) GOTO 760
C
C     CALCULATIONS FOR ACTUATOR CONFIG. 1 AND 2:
      YHAT(K-1) = A1(K-1)*YHAT(K-1)+A2(K-1)*YHAT(K-2)
     *      +B1(K-1)*UC(K-2)+B2(K-1)*UC(K-3)
      GAM1(K-1) = UC(K-2)+A1(K-1)*GAM1(K-2)+A2(K-1)*GAM1(K-3)
      GAM2(K-1) = UC(K-3)+A1(K-1)*GAM2(K-2)+A2(K-1)*GAM2(K-3)
      GOTO 770
C
C     CALCULATIONS FOR ACTUATOR CONFIG. 3 AND 4:
  760 YHAT(K-1) = A1(K-1)*YHAT(K-1)+A2(K-1)*YHAT(K-2)
     *           +B1(K-1)*U(K-2)+B2(K-1)*U(K-3)
      GAM1(K-1) = U(K-2)+A1(K-1)*GAM1(K-2)+A2(K-1)*GAM1(K-3)
      GAM2(K-1) = U(K-3)+A1(K-1)*GAM2(K-2)+A2(K-1)*GAM2(K-3)
  770 LAM1(K-1) = YHAT(K-2)+A1(K-1)*LAM1(K-2)+A2(K-1)*LAM1(K-3)
      LAM2(K-1) = YHAT(K-3)+A1(K-1)*LAM2(K-2)+A2(K-1)*LAM2(K-3)
      E(K-1) = Y(K-1)-YHAT(K-1)
      DENOM  = H+MU(1)*LAM1(K-1)**2+MU(2)*LAM2(K-1)**2
     *           +RHO(1)*GAM1(K-1)**2+RHO(2)*GAM2(K-1)**2
      IF(E(K-1).LT.1.0D-10) E(K-1) = 0.0D0
      IF(DENOM.GT.1.0D05) DENOM = 1.0D05
      V(K-1) = E(K-1)/DENOM
C
C     CALCULATE NEW PLANT PARAMETER ESTIMATES
      A1(K) = A1(K-1)+MU(1)*V(K-1)*LAM1(K-1)
      A2(K) = A2(K-1)+MU(2)*V(K-1)*LAM2(K-1)
      B1(K) = B1(K-1)+RHO(1)*V(K-1)*GAM1(K-1)
      B2(K) = B2(K-1)+RHO(2)*V(K-1)*GAM2(K-1)
C
C     CALCULATE NEW CONTROLLER PARAMETERS
      CALL CNTSET(GAMMA)
      RETURN
      END

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     SOI ALGORITHM
C
C     EQUATIONS (4-76) - (4-81)
C
      SUBROUTINE SOI(GAMMA,MU,RHO,H,Q)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 R(5),U(5),Y(5),S(5),YHAT(5),UC(5)
      REAL*8 GAMMA(2),MU(5),RHO(5),Q(2)
      REAL*8 A1(5),A2(5),B1(5),B2(5)
      REAL*8 V(5),Z(5),ETA1(5),ETA2(5),NU1(5),NU2(5)
      COMMON /SIGNAL/R,U,Y,S,UC
      COMMON /CONTRL/A1,A2,B1,B2,ETA1,ETA2,NU1,NU2,NACT
      COMMON /MISC2/YHAT,Z,V
      DATA K/5/
C
C     SHIFT VARIABLES TO ALLOW CALCULATION OF NEW VALUES
      Z(K-3) = Z(K-2)
      Z(K-2) = Z(K-1)
      Z(K-1) = Z(K)
      A1(K-2) = A1(K-1)
      A1(K-1) = A1(K)
```

```
      A2(K-2)  = A2(K-1)
      A2(K-1)  = A2(K)
      B1(K-2)  = B1(K-1)
      B1(K-1)  = B1(K)
      B2(K-2)  = B2(K-1)
      B2(K-1)  = B2(K)
      YHAT(K-2) = YHAT(K-1)
      YHAT(K-1) = YHAT(K)
      V(K-2)  = V(K-1)
      V(K-1)  = V(K)
C
      IF(NACT.GT.2) GOTO 660
C
C     CALCULATIONS FOR ACTUATOR CONFIG. 1 AND 2:
      YHAT(K-1) = A1(K-1)*Z(K-2)+A2(K-1)*Z(K-3)
     *      +B1(K-1)*UC(K-2)+B2(K-1)*UC(K-3)
      DENOM = H+MU(1)*Z(K-2)**2+MU(2)*Z(K-3)**2
     *      +RHO(1)*UC(K-2)**2+RHO(2)*UC(K-3)**2
      GOTO 670
C
C     CALCULATIONS FOR ACTUATOR CONFIG. 3 AND 4
  660 YHAT(K-1) = A1(K-1)*Z(K-2)+A2(K-1)*Z(K-3)
     *          +B1(K-1)*U(K-2)+B2(K-1)*U(K-3)
      DENOM   = H+MU(1)*Z(K-2)**2+MU(2)*Z(K-3)**2
     *          +RHO(1)*U(K-2)**2+RHO(2)*U(K-3)**2
  670 V(K-1) = (Y(K-1)-YHAT(K-1)+G(1)*(Y(K-2)-Z(K-2))
     *          +G(2)*(Y(K-3)-Z(K-3)))/DENOM
C
C     CALCULATE NEW PLANT PARAMETER ESTIMATES
      A1(K)   = A1(K-1)+MU(1)*V(K-1)*Z(K-2)
      A2(K)   = A2(K-1)+MU(2)*V(K-1)*Z(K-3)
      B1(K)   = B1(K-1)+RHO(1)*V(K-1)*U(K-2)
      B2(K)   = B2(K-1)+RHO(2)*V(K-1)*U(K-3)
C
      IF(NACT.GT.2) GOTO 680
      Z(K-1) = A1(K)*Z(K-2)+A2(K)*Z(K-3)+B1(K)*UC(K-2)+B2(K)*UC(K-3)
      GOTO 690
C
  680 Z(K-1) = A1(K)*Z(K-2)+A2(K)*Z(K-3)
     *         +B1(K)*U(K-2)+B2(K)*U(K-3)
  690 CONTINUE
C
C     CALCULATE NEW CONTROLLER PARAMETERS
      CALL CNTSET(GAMMA)
      RETURN
      END


      SUBROUTINE STAT2(N,KOUNT)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 R(5),U(5),Y(5),S(5),UC(5)
      REAL*8 IDD(10),ID(10),NITE(10),SAS1(10),
     *  SAPEA1(10),SAPEA2(10),SAPEB1(10),SAPEB2(10),
     *  SPEVA1(10),SPEVA2(10),SPEVB1(10),SPEVB2(10),
     *  SACEE1(10),SACEE2(10),SACEN1(10),SACEN2(10),
     *  SCEVE1(10),SCEVE2(10),SCEVN1(10),SCEVN2(10),
     *  NASTE(10),NSTEV(10)
      REAL*8 A1(5),A2(5),B1(5),B2(5),ETA1(5),ETA2(5),NU1(5),NU2(5)
      INTEGER KD(12)
      COMMON /STATS/IDD,ID,NITE,SAS1,SAPEA1,SAPEA2,SAPEB1,SAPEB2,
     *  SPEVA1,SPEVA2,SPEVB1,SPEVB2,SACEE1,SACEE2,SACEN1,SACEN2,
     *  SCEVE1,SCEVE2,SCEVN1,SCEVN2,
     *  NASTE,NSTEV,STE,
     *  SSPA1,SSPA2,SSPB1,SSPB2,SSCE1,SSCE2,SSCN1,SSCN2,M,KD
      COMMON /CONTRL/A1,A2,B1,B2,ETA1,ETA2,NU1,NU2,NACT
      COMMON /SIGNAL/R,U,Y,S,UC
      DATA K/5/
C
      IF(DABS(S(K)).LT.1.0D-03) GOTO 100
      TE = (S(K)-Y(K))/S(K)
      GOTO 200
  100 TE = 0.0
      KOUNT = KOUNT+1
  200 NASTE(M) = NASTE(M)+TE**2
      STE = STE+TE
      SAS1(M)   = SAS1(M)+UC(K-1)**2
      SAPEA1(M) = SAPEA1(M)+A1(K-1)
      SAPEA2(M) = SAPEA2(M)+A2(K-1)
      SAPEB1(M) = SAPEB1(M)+B1(K-1)
      SAPEB2(M) = SAPEB2(M)+B2(K-1)
```

```
      SACEE1(M) = SACEE1(M)+ETA1(K-1)
      SACEE2(M) = SACEE2(M)+ETA2(K-1)
      SACEN1(M) = SACEN1(M)+NU1(K-1)
      SACEN2(M) = SACEN2(M)+NU2(K-1)
      SSPA1 = SSPA1+A1(K-1)**2
      SSPA2 = SSPA2+A2(K-1)**2
      SSPB1 = SSPB1+B1(K-1)**2
      SSPB2 = SSPB2+B2(K-1)**2
      SSCE1 = SSCE1+ETA1(K-1)**2
      SSCE2 = SSCE2+ETA2(K-1)**2
      SSCN1 = SSCN1+NU1(K-1)**2
      SSCN2 = SSCN2+NU2(K-1)**2
C
  300 TIME = FLOAT(N)
      IF(N.LT.KD(M+1)) RETURN
      IDD(M) = S(K)
      ID(M)  = Y(K)
      RANGE  = FLOAT(KD(M+1)-KD(M))
      DIV = (RANGE-FLOAT(KOUNT))*(RANGE-FLOAT(KOUNT)-1)
      RNG = RANGE-FLOAT(KOUNT)
      NSTEV(M)  = 0.0
      IF(DIV.GT.0.0) NSTEV(M) = (RNG*NASTE(M)-STE**2)/DIV
      IF(NSTEV(M).LT.0.0D0) NSTEV(M) = 0.0D0
      NITE(M) = TE
      DIVSOR = RANGE*(RANGE-1)
      IF(DIVSOR.LE.0.0) GOTO 1000
      SPEVA1(M) = (RANGE*SSPA1-SAPEA1(M)**2)/DIVSOR
      IF(SPEVA1(M).LT.0.0D0) SPEVA1(M) = 0.0D0
      SPEVA2(M) = (RANGE*SSPA2-SAPEA2(M)**2)/DIVSOR
      IF(SPEVA2(M).LT.0.0D0) SPEVA2(M) = 0.0D0
      SPEVB1(M) = (RANGE*SSPB1-SAPEB1(M)**2)/DIVSOR
      IF(SPEVB1(M).LT.0.0D0) SPEVB1(M) = 0.0D0
      SPEVB2(M) = (RANGE*SSPB2-SAPEB2(M)**2)/DIVSOR
      IF(SPEVB2(M).LT.0.0D0) SPEVB2(M) = 0.0D0
C
      SCEVE1(M) = (RANGE*SSCE1-SACEE1(M)**2)/DIVSOR
      IF(SCEVE1(M).LT.0.0D0) SCEVE1(M) = 0.0D0
      SCEVE2(M) = (RANGE*SSCE2-SACEE2(M)**2)/DIVSOR
      IF(SCEVE2(M).LT.0.0D0) SCEVE2(M) = 0.0D0
      SCEVN1(M) = (RANGE*SSCN1-SACEN1(M)**2)/DIVSOR
      IF(SCEVN1(M).LT.0.0D0) SCEVN1(M) = 0.0D0
      SCEVN2(M) = (RANGE*SSCN2-SACEN2(M)**2)/DIVSOR
      IF(SCEVN2(M).LT.0.0D0) SCEVN2(M) = 0.0D0
      GOTO 2000
 1000 SPEVA1(M) = 0.0D0
      SPEVA2(M) = 0.0D0
      SPEVB1(M) = 0.0D0
      SPEVB2(M) = 0.0D0
      SCEVE1(M) = 0.0D0
      SCEVE2(M) = 0.0D0
      SCEVN1(M) = 0.0D0
      SCEVN2(M) = 0.0D0
 2000 CONTINUE
      SAS1(M)   = SAS1(M)/RANGE
      SAPEA1(M) = SAPEA1(M)/RANGE
      SAPEA2(M) = SAPEA2(M)/RANGE
      SAPEB1(M) = SAPEB1(M)/RANGE
      SAPEB2(M) = SAPEB2(M)/RANGE
      SACEE1(M) = SACEE1(M)/RANGE
      SACEE2(M) = SACEE2(M)/RANGE
      SACEN1(M) = SACEN1(M)/RANGE
      SACEN2(M) = SACEN2(M)/RANGE
      NASTE(M) = NASTE(M)/RANGE
      KOUNT = 0
      STE = 0.0D0
      SSPA1 = 0.0D0
      SSPA2 = 0.0D0
      SSPB1 = 0.0D0
      SSPB2 = 0.0D0
      SSCE1 = 0.0D0
      SSCE2 = 0.0D0
      SSCN1 = 0.0D0
      SSCN2 = 0.0D0
      M = M+1
      RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     SUBROUTINE - CONTROLLER SETTINGS
C
C     EQUATIONS (4-89) TO (4-92)
C
      SUBROUTINE CNTSET(GAMMA)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 A1(5),A2(5),B1(5),B2(5)
      REAL*8 ETA1(5),ETA2(5),NU1(5),NU2(5)
      REAL*8 GAMMA(2)
C
      COMMON /CONTRL/A1,A2,B1,B2,ETA1,ETA2,NU1,NU2,NACT
      DATA K/5/
C
C     CALCULATE CONTROLLER COEFFICIENTS
      ETA1(K-1) = GAMMA(1)-A1(K)
      NU1(K-1)  = (ETA1(K-1)*A2(K)+(ETA1(K-1)*A1(K)-A2(K)+
     *            GAMMA(2))*(A1(K)-A2(K)*B1(K)/B2(K)))/
     *            (B1(K)*A1(K)+B2(K)-A2(K)*B1(K)**2/B2(K))
      ETA2(K-1) = A1(K)*ETA1(K-1)+GAMMA(2)-B1(K)*NU1(K-1)-A2(K)
      NU2(K-1)  = ETA2(K-1)*A2(K)/B2(K)
C
      RETURN
      END
```